53-1002603-01
28 September 2012

Brocade ICX 6650

Layer 3 Routing Configuration Guide

Supporting FastIron Software Release 07.5.00

BROCADE

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. shall have no liability or responsibility to any person or entity with respect to any loss, cost, liability, or damages arising from the information contained in this book or the computer programs that accompany it.

The product described by this document may contain “open source” software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit *http://www.brocade.com/support/oscd*.

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Brocade ICX 6650 Layer 3 Routing Configuration Guide	53-1002603-01	Release 07.4.00 document updated with enhancements in Release 07.5.00	September 2012

Contents

About This Document

Chapter 1 IP Configuration

Chapter 2 Base Layer 3 and Routing Protocols

Chapter 3 RIP (IPv4)

Chapter 4 RIP (IPv6)

Chapter 6 OSPF version 3 (IPv6)

Chapter 7 BGP (IPv4)

Chapter 8 IPv6

Chapter 9 VRRP and VRRP-E

About This Document

The Brocade ICX 6650 is a ToR (Top of Rack) Ethernet switch for campus LAN and classic Ethernet data center environments.

Audience

This document is designed for system administrators with a working knowledge of Layer 2 and Layer 3 switching and routing.

If you are using a Brocade Layer 3 Switch, you should be familiar with the following protocols if applicable to your network: IP, RIP, OSPF, BGP, ISIS, PIM, and VRRP.

Supported hardware and software

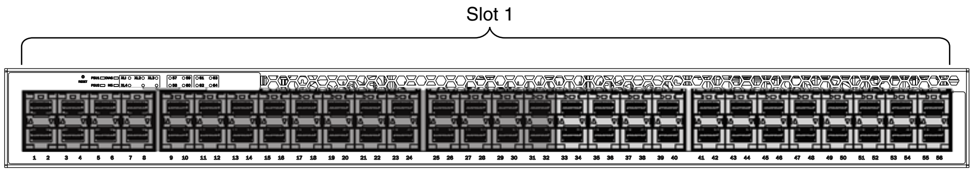
This document is specific to the Brocade ICX 6650 running FastIron 7.5.00.

Brocade ICX 6650 slot and port numbering

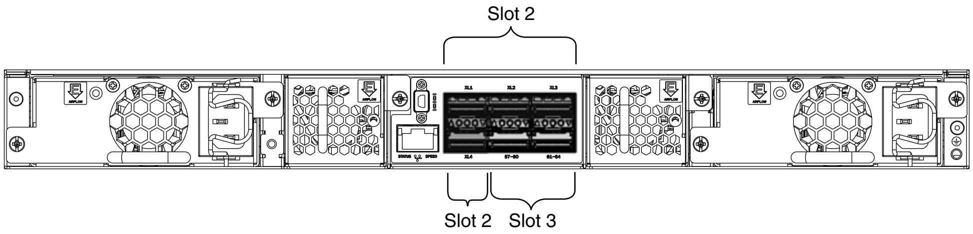
Many CLI commands require users to enter port numbers as part of the command syntax, and many **show** command outputs display port numbers. The port numbers are entered and displayed in stack-unit/slot number/port number format. In all Brocade ICX 6650 inputs and outputs, the stack-unit number is always 1.

The Brocade ICX 6650 contains the following slots and Ethernet ports:

- Slot 1 is located on the front of the ICX 6650 device and contains ports 1 through 56. Ports 1 through 32 are 10 GbE. Ports 33 through 56 are 1/10 GbE SFP+ ports. Refer to the following figure.



- Slot 2 is located on the back of the Brocade ICX 6650 device and contains ports 1 through 3 on the top row and port 4 on the bottom row. These ports are 2x40 GbE QSFP+. Refer to the following figure.



- Slot 3 is located on the back of the Brocade ICX 6650 device and contains ports 1 through 8. These ports are 4 x 10 GbE breakout ports and require the use of a breakout cable. Refer to the previous figure.

How this document is organized

This document is organized to help you find the information that you want as quickly and easily as possible.

The document contains the following components:

- "IP Configuration" on page 1
- "Base Layer 3 and Routing Protocols" on page 133
- "RIP (IPv4)" on page 141
- "RIP (IPv6)" on page 157
- "OSPF version 2 (IPv4)" on page 167
- "OSPF version 3 (IPv6)" on page 227
- "BGP (IPv4)" on page 281
- "IPv6" on page 401
- "VRRP and VRRP-E" on page 411

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names Identifies the names of user-manipulated GUI elements Identifies keywords and operands Identifies text to enter at the GUI or CLI
italic text	Provides emphasis Identifies variables Identifies paths and Internet addresses Identifies document titles
`code` text	Identifies CLI output Identifies command syntax examples

For readability, command names in the narrative portions of this guide are presented in mixed lettercase: for example, **switchShow**. In actual examples, command lettercase is all lowercase.

Command syntax conventions

Command syntax in this manual follows these conventions:

command	Commands are printed in bold.
--**option, option**	Command options are printed in bold.
-**argument,** arg	Arguments.
[ ]	Optional elements appear in brackets.
variable	Variables are printed in italics. In the help pages, values are underlined or enclosed in angled brackets < >.
...	Repeat the previous element, for example "member[;member...]"
value	Fixed values following arguments are printed in plain font. For example, --**show** WWN
|	Boolean. Elements are exclusive. Example: --**show** -**mode** egress | ingress

Notes, cautions, and warnings

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE
A note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates potential damage to hardware or data.

CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER

A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Notice to the reader

This document might contain references to the trademarks of the following corporations. These trademarks are the properties of their respective companies and corporations.

These references are made for informational purposes only.

Corporation	Referenced Trademarks and Products
Microsoft Corporation	Windows, Windows NT, Internet Explorer
Oracle Corporation	Oracle, Java
Netscape Communications Corporation	Netscape
Mozilla Corporation	Mozilla Firefox
Sun Microsystems, Inc.	Sun, Solaris
Red Hat, Inc.	Red Hat, Red Hat Network, Maximum RPM, Linux Undercover

Related publications

The following Brocade documents supplement the information in this guide:

- *Brocade ICX 6650 Release Notes*
- *Brocade ICX 6650 Hardware Installation Guide New*
- *Brocade ICX 6650 Administration Guide*
- *Brocade ICX 6650 Platform and Layer 2 Configuration Guide*
- *Brocade ICX 6650 Layer 3 Routing Configuration Guide*
- *Brocade ICX 6650 Security Configuration Guide*
- *Brocade ICX 6650 IP Multicast Configuration Guide*

- *Brocade ICX 6650 Diagnostic Reference*
- *Unified IP MIB Reference*
- *Ports-on-Demand Licensing for the Brocade ICX 6650*

The latest versions of these guides are posted at *http://www.brocade.com/ethernetproducts*.

Additional information

This section lists additional Brocade and industry-specific documentation that you might find helpful.

Brocade resources

To get up-to-the-minute information, go to *http://my.brocade.com* to register at no cost for a user ID and password.

White papers, online demonstrations, and data sheets are available through the Brocade website at:

http://www.brocade.com/products-solutions/products/index.page

For additional Brocade documentation, visit the Brocade website:

http://www.brocade.com

Release notes are available on the MyBrocade website.

Other industry resources

For additional resource information, visit the Technical Committee T11 website. This website provides interface standards for high-performance and mass storage applications for Fibre Channel, storage management, and other applications:

http://www.t11.org

For information about the Fibre Channel industry, visit the Fibre Channel Industry Association website:

http://www.fibrechannel.org

Getting technical help

To contact Technical Support, go to

http://www.brocade.com/services-support/index.page

for the latest e-mail and telephone contact information.

Brocade ICX 6650 slot and port numbering

Document feedback

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. Forward your feedback to:

documentation@brocade.com

Provide the title and version number of the document and as much detail as possible about your comment, including the topic heading and page number and your suggestions for improvement.

Chapter

IP Configuration

1

Table 1 lists the IP features Brocade ICX 6650 devices support. These features are supported with the full Layer 3 software image, except where explicitly noted.

TABLE 1 Supported IP features

Feature	Brocade ICX 6650
BootP/DHCP relay	Yes
Specifying which IP address will be included in a DHCP/BootP reply packet	Yes
DHCP Server	Yes
DHCP Client-Based Auto-Configuration	Yes
DHCP Client-Based Flash image Auto-update	Yes
DHCP assist	Yes
Equal Cost Multi Path (ECMP) load sharing	Yes
IP helper	Yes
Single source address for the following packet types: • Telnet • TFTP • Syslog • SNTP • TACACS/TACACS+ • RADIUS • SSH • SNMP	Yes
IPv4 point-to-point GRE IP tunnels	Yes
Routes in hardware maximum: Up to 7168 routes	Yes
Routing for directly connected IP subnets	Yes
Virtual Interfaces: Up to 512 virtual interfaces	Yes
31-bit subnet mask on point-to-point networks	Yes
Address Resolution Protocol (ARP)	Yes
Reverse Address Resolution Protocol (RARP)	Yes
IP follow	Yes
Proxy ARP	Yes

TABLE 1 Supported IP features (Continued)

Feature	Brocade ICX 6650
Local proxy ARP	Yes
Jumbo frames • Up to 10,240 bytes	Yes
IP MTU (individual port setting)	Yes
Path MTU discovery	Yes
ICMP Router Discovery Protocol (IRDP)	Yes
Domain Name Server (DNS) resolver	Yes

NOTE
The terms Layer 3 Switch and router are used interchangeably in this chapter and mean the same.

Basic IP configuration

IP is enabled by default. Basic configuration consists of adding IP addresses for Layer 3 Switches, enabling a route exchange protocol, such as the Routing Information Protocol (RIP).

If you are configuring a Layer 3 Switch, refer to "Configuring IP addresses" on page 19 to add IP addresses, then enable and configure the route exchange protocols, as described in other chapters of this guide.

If you are configuring a Layer 2 Switch, refer to "Configuring the management IP address and specifying the default gateway" on page 88 to add an IP address for management access through the network and to specify the default gateway.

The rest of this chapter describes IP and how to configure it in more detail. Use the information in this chapter if you need to change some of the IP parameters from their default values or you want to view configuration information or statistics.

IP configuration overview

Brocade Layer 2 Switches and Layer 3 Switches support Internet Protocol version 4 (IPv4) and IPv6. IP support on Brocade Layer 2 Switches consists of basic services to support management access and access to a default gateway.

Full Layer 3 support

IP support on Brocade full Layer 3 Switches includes all of the following, in addition to a highly configurable implementation of basic IP services including Address Resolution Protocol (ARP), ICMP Router Discovery Protocol (IRDP), and Reverse ARP (RARP):

- Route-only support (Global configuration level only)
- Route redistribution

- Route exchange protocols:
  - Routing Information Protocol (RIP)
  - Open Shortest Path First (OSPF)
  - Border Gateway Protocol version 4 (BGP4)
- Multicast protocols:
  - Internet Group Membership Protocol (IGMP)
  - Protocol Independent Multicast Dense (PIM-DM)
  - Protocol Independent Multicast Sparse (PIM-SM)
- Router redundancy protocols:
  - Virtual Router Redundancy Protocol Extended (VRRP-E)
  - Virtual Router Redundancy Protocol (VRRP)

IP interfaces

NOTE
This section describes IPv4 addresses. For information about IPv6 addresses on Brocade ICX 6650 devices, refer to “IPv6 addressing overview” section in the *Brocade ICX 6650 Administration Guide*.

Brocade Layer 3 Switches and Layer 2 Switches allow you to configure IP addresses. On Layer 3 Switches, IP addresses are associated with individual interfaces. On Layer 2 Switches, a single IP address serves as the management access address for the entire device.

All Brocade Layer 3 Switches and Layer 2 Switches support configuration and display of IP addresses in classical subnet format (for example: 192.168.1.1 255.255.255.0) and Classless Interdomain Routing (CIDR) format (for example: 192.168.1.1/24). You can use either format when configuring IP address information. IP addresses are displayed in classical subnet format by default but you can change the display format to CIDR. Refer to “Changing the network mask display to prefix format” on page 113.

Layer 3 Switches

Brocade Layer 3 Switches allow you to configure IP addresses on the following types of interfaces:

- Ethernet ports
- Virtual routing interfaces (used by VLANs to route among one another)
- Loopback interfaces

Each IP address on a Layer 3 Switch must be in a different subnet. You can have only one interface that is in a given subnet. For example, you can configure IP addresses 192.168.1.1/24 and 192.168.2.1/24 on the same Layer 3 Switch, but you cannot configure 192.168.1.1/24 and 192.168.1.2/24 on the same Layer 3 Switch.

You can configure multiple IP addresses on the same interface.

The number of IP addresses you can configure on an individual interface depends on the Layer 3 Switch model. To display the maximum number of IP addresses and other system parameters you can configure on a Layer 3 Switch, refer to “Displaying and modifying system parameter default settings” section in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

You can use any of the IP addresses you configure on the Layer 3 Switch for Telnet, or SNMP access.

Layer 2 Switches

You can configure an IP address on a Brocade Layer 2 Switch for management access to the Layer 2 Switch. An IP address is required for Telnet access and SNMP access.

You also can specify the default gateway for forwarding traffic to other subnets.

IP packet flow through a Layer 3 Switch

Figure 1 shows how an IP packet moves through a Brocade Layer 3 Switch.

FIGURE 1 IP Packet flow through a Brocade Layer 3 Switch

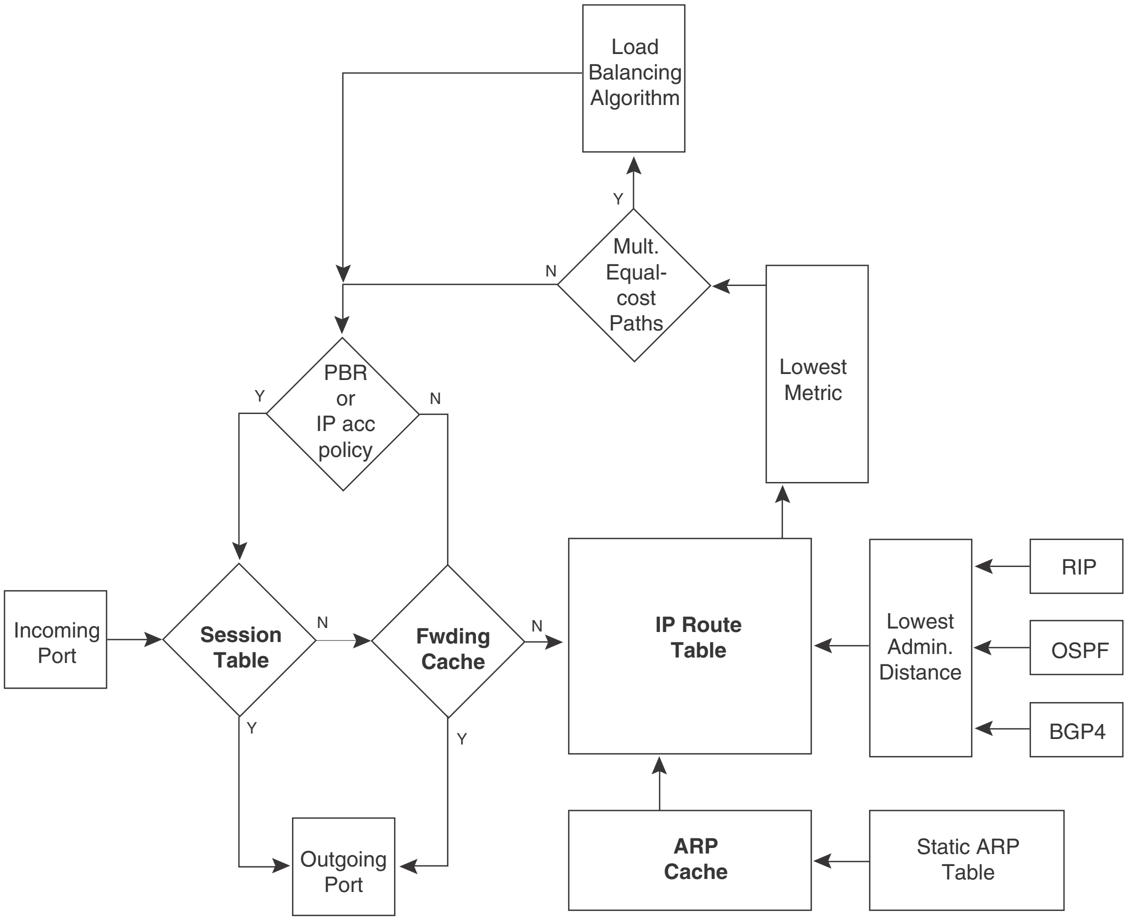


Figure 1 shows the following packet flow:

1. When the Layer 3 Switch receives an IP packet, the Layer 3 Switch checks for filters on the receiving interface.[1] If a deny filter on the interface denies the packet, the Layer 3 Switch discards the packet and performs no further processing, except generating a Syslog entry and SNMP message, if logging is enabled for the filter.
2. If the packet is not denied at the incoming interface, the Layer 3 Switch looks in the session table for an entry that has the same source IP address and TCP or UDP port as the packet. If the session table contains a matching entry, the Layer 3 Switch immediately forwards the packet, by addressing it to the destination IP address and TCP or UDP port listed in the session table entry and sending the packet to a queue on the outgoing ports listed in the session table. The Layer 3 Switch selects the queue based on the Quality of Service (QoS) level associated with the session table entry.

[1] The filter can be an Access Control List (ACL) or an IP access policy.

3. If the session table does not contain an entry that matches the packet source address and TCP or UDP port, the Layer 3 Switch looks in the IP forwarding cache for an entry that matches the packet destination IP address. If the forwarding cache contains a matching entry, the Layer 3 Switch forwards the packet to the IP address in the entry. The Layer 3 Switch sends the packet to a queue on the outgoing ports listed in the forwarding cache. The Layer 3 Switch selects the queue based on the Quality of Service (QoS) level associated with the forwarding cache entry.
4. If the IP forwarding cache does not have an entry for the packet, the Layer 3 Switch checks the IP route table for a route to the packet destination. If the IP route table has a route, the Layer 3 Switch makes an entry in the session table or the forwarding cache, and sends the route to a queue on the outgoing ports:
   - If the running-config contains an IP access policy for the packet, the software makes an entry in the session table. The Layer 3 Switch uses the new session table entry to forward subsequent packets from the same source to the same destination.
   - If the running-config does not contain an IP access policy for the packet, the software creates a new entry in the forwarding cache. The Layer 3 Switch uses the new cache entry to forward subsequent packets to the same destination.

The following sections describe the IP tables and caches:

- ARP cache and static ARP table
- IP route table
- IP forwarding cache
- Layer 4 session table

The software enables you to display these tables. You also can change the capacity of the tables on an individual basis if needed by changing the memory allocation for the table.

ARP cache and static ARP table

The ARP cache contains entries that map IP addresses to MAC addresses. Generally, the entries are for devices that are directly attached to the Layer 3 Switch.

An exception is an ARP entry for an interface-based static IP route that goes to a destination that is one or more router hops away. For this type of entry, the MAC address is either the destination device MAC address or the MAC address of the router interface that answered an ARP request on behalf of the device, using proxy ARP.

ARP cache

The ARP cache can contain dynamic (learned) entries and static (user-configured) entries. The software places a dynamic entry in the ARP cache when the Layer 3 Switch learns a device MAC address from an ARP request or ARP reply from the device.

The software can learn an entry when the Layer 2 Switch or Layer 3 Switch receives an ARP request from another IP forwarding device or an ARP reply. Here is an example of a dynamic entry:

```
      IP Address         MAC Address         Type        Age      Port
1     10.95.6.102        0000.00fc.ea21      Dynamic     0         1/1/6
```

Each entry contains the destination device IP address and MAC address.

Static ARP table

In addition to the ARP cache, Layer 3 Switches have a static ARP table. Entries in the static ARP table are user-configured. You can add entries to the static ARP table regardless of whether or not the device the entry is for is connected to the Layer 3 Switch.

NOTE
Layer 3 Switches have a static ARP table. Layer 2 Switches do not.

The software places an entry from the static ARP table into the ARP cache when the entry interface comes up.

Here is an example of a static ARP entry.

```
No.    IP Address        MAC Address     Type    Age    Port     Status
1      192.168.6.111   0000.003b.d210 Static    0     1/1/1    Valid
```

Each entry lists the information you specified when you created the entry.

Displaying ARP entries

To display ARP entries, refer to the following sections:

- "Displaying the ARP cache" on page 118 – Layer 3 Switch
- "Displaying the static ARP table" on page 120 – Layer 3 Switch only
- "Displaying ARP entries" on page 129 – Layer 2 Switch

To configure other ARP parameters, refer to the following sections:

- "ARP parameter configuration" on page 35 – Layer 3 Switch only

To increase the size of the ARP cache and static ARP table, refer to the following:

- For dynamic entries, refer to the section "Displaying and modifying system parameter default settings" section in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*. The *ip-arp* parameter controls the ARP cache size.
- Static entries, "Changing the maximum number of entries the static ARP table can hold" on page 40 (Layer 3 Switches only). The *ip-static-arp* parameter controls the static ARP table size.

IP route table

The IP route table contains paths to IP destinations.

NOTE
Layer 2 Switches do not have an IP route table. A Layer 2 Switch sends all packets addressed to another subnet to the default gateway, which you specify when you configure the basic IP information on the Layer 2 Switch.

The IP route table can receive the paths from the following sources:

- A directly-connected destination, which means there are no router hops to the destination
- A static IP route, which is a user-configured route
- A route learned through RIP
- A route learned through OSPF
- A route learned through BGP4

The IP route table contains the best path to a destination:

- When the software receives paths from more than one of the sources listed above, the software compares the administrative distance of each path and selects the path with the lowest administrative distance. The administrative distance is a protocol-independent value from 1 through 255.
- When the software receives two or more best paths from the same source and the paths have the same metric (cost), the software can load share traffic among the paths based on destination host or network address (based on the configuration and the Layer 3 Switch model).

Here is an example of an entry in the IP route table.

```
Destination       NetMask           Gateway           Port   Cost    Type
10.1.0.0           255.255.0.0       10.1.1.2          1/1/1   2       R
```

Each IP route table entry contains the destination IP address and subnet mask and the IP address of the next-hop router interface to the destination. Each entry also indicates the port attached to the destination or the next-hop to the destination, the route IP metric (cost), and the type. The type indicates how the IP route table received the route:

- To display the IP route table, refer to "Displaying the IP route table" on page 122 (Layer 3 Switch only).
- To configure a static IP route, refer to "Static routes configuration" on page 45 (Layer 3 Switch only).
- To clear a route from the IP route table, refer to "Clearing IP routes" on page 124 (Layer 3 Switch only).
- To increase the size of the IP route table for learned and static routes, refer to the section "Displaying and modifying system parameter default settings" section in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*:
  - For learned routes, modify the *ip-route* parameter.
  - For static routes, modify the *ip-static-route* parameter.

IP forwarding cache

The IP forwarding cache provides a fast-path mechanism for forwarding IP packets. The cache contains entries for IP destinations. When a Brocade Layer 3 Switch has completed processing and addressing for a packet and is ready to forward the packet, the device checks the IP forwarding cache for an entry to the packet destination:

- If the cache contains an entry with the destination IP address, the device uses the information in the entry to forward the packet out the ports listed in the entry. The destination IP address is the address of the packet final destination. The port numbers are the ports through which the destination can be reached.
- If the cache does not contain an entry and the traffic does not qualify for an entry in the session table instead, the software can create an entry in the forwarding cache.

Each entry in the IP forwarding cache has an age timer. If the entry remains unused for ten minutes, the software removes the entry. The age timer is not configurable.

Here is an example of an entry in the IP forwarding cache.

```
      IP Address       Next Hop         MAC              Type  Port  Vlan  Pri
1     192.168.1.11     DIRECT           0000.0000.0000   PU    n/a         0
```

Each IP forwarding cache entry contains the IP address of the destination, and the IP address and MAC address of the next-hop router interface to the destination. If the destination is actually an interface configured on the Layer 3 Switch itself, as shown here, then next-hop information indicates this. The port through which the destination is reached is also listed, as well as the VLAN and Layer 4 QoS priority associated with the destination if applicable.

To display the IP forwarding cache, refer to “Displaying the forwarding cache” on page 121.

NOTE
You cannot add static entries to the IP forwarding cache, although you can increase the number of entries the cache can contain. Refer to the section “Displaying and modifying system parameter default settings” section in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

Layer 4 session table

The Layer 4 session provides a fast path for forwarding packets. A ***session*** is an entry that contains complete Layer 3 and Layer 4 information for a flow of traffic. Layer 3 information includes the source and destination IP addresses. Layer 4 information includes the source and destination TCP and UDP ports. For comparison, the IP forwarding cache contains the Layer 3 destination address but does not contain the other source and destination address information of a Layer 4 session table entry.

The Layer 2 Switch or Layer 3 Switch selects the session table instead of the IP forwarding table for fast-path forwarding for the following features:

- Layer 4 Quality-of-Service (QoS) policies
- IP access policies

To increase the size of the session table, refer to the section “Displaying and modifying system parameter default settings” section in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*. The ip-qos-session parameter controls the size of the session table.

IP route exchange protocols

Brocade Layer 3 Switches support the following IP route exchange protocols:

- Routing Information Protocol (RIP)
- Open Shortest Path First (OSPF)
- Border Gateway Protocol version 4 (BGP4)

All these protocols provide routes to the IP route table. You can use one or more of these protocols, in any combination. The protocols are disabled by default. For configuration information, refer to the following:

- Chapter 3, “RIP (IPv4)”
- Chapter 5, “OSPF version 2 (IPv4)”
- Chapter 7, “BGP (IPv4)”

IP multicast protocols

Brocade Layer 3 Switches also support the following Internet Group Membership Protocol (IGMP) based IP multicast protocols:

- Protocol Independent Multicast – Dense mode (PIM-DM)
- Protocol Independent Multicast – Sparse mode (PIM-SM)

For configuration information, refer to the *Brocade ICX 6650 IP Multicast Configuration Guide*. .

NOTE
Brocade Layer 2 Switches support IGMP and can forward IP multicast packets. For more information see, Chapter 2, "IP Multicast Reduction" in the *Brocade ICX 6650 IP Mulitcast Configuration Guide*.

IP interface redundancy protocols

You can configure a Brocade Layer 3 Switch to back up an IP interface configured on another Brocade Layer 3 Switch. If the link for the backed up interface becomes unavailable, the other Layer 3 Switch can continue service for the interface. This feature is especially useful for providing a backup to a network default gateway.

Brocade Layer 3 Switches support the following IP interface redundancy protocols:

- Virtual Router Redundancy Protocol (VRRP) – A standard router redundancy protocol based on RFC 2338. You can use VRRP to configure Brocade Layer 3 Switches and third-party routers to back up IP interfaces on other Brocade Layer 3 Switches or third-party routers.
- Virtual Router Redundancy Protocol Extended (VRRP-E) – A Brocade extension to standard VRRP that adds additional features and overcomes limitations in standard VRRP. You can use VRRP-E only on Brocade Layer 3 Switches.

For configuration information, refer to the Chapter 9, "VRRP and VRRP-E".

ACLs and IP access policies

Brocade Layer 3 Switches provide two mechanisms for filtering IP traffic:

- Access Control Lists (ACLs)
- IP access policies

Both methods allow you to filter packets based on Layer 3 and Layer 4 source and destination information.

ACLs also provide great flexibility by providing the input to various other filtering mechanisms such as route maps, which are used by BGP4.

IP access policies allow you to configure QoS based on sessions (Layer 4 traffic flows).

Only one of these filtering mechanisms can be enabled on a Brocade device at a time. Brocade devices can store forwarding information for both methods of filtering in the session table.

For configuration information, see the Chapter, "Rule-Based IP ACLs" in the *Brocade ICX 6650 Security Configuration Guide*.

Basic IP parameters and defaults – Layer 3 Switches

IP is enabled by default. The following IP-based protocols are all disabled by default:

- Routing protocols:
  - Routing Information Protocol (RIP) – refer to Chapter 3, "RIP (IPv4)"
  - Open Shortest Path First (OSPF) – refer to Chapter 5, "OSPF version 2 (IPv4)"
  - Border Gateway Protocol version 4 (BGP4) – refer to Chapter 7, "BGP (IPv4)"
- Multicast protocols:
  - Internet Group Membership Protocol (IGMP)
  - Protocol Independent Multicast Dense (PIM-DM)
  - Protocol Independent Multicast Sparse (PIM-SM)

  NOTE
  For more information, see the *Brocade ICX 6650 IP Mulitcast Configuration Guide*.

- Router redundancy protocols:
  - Virtual Router Redundancy Protocol Extended (VRRP-E) – refer to Chapter 9, "VRRP and VRRP-E"
  - Virtual Router Redundancy Protocol (VRRP) – refer to Chapter 9, "VRRP and VRRP-E"

The following tables list the Layer 3 Switch IP parameters, their default values, and where to find configuration information.

NOTE
For information about parameters in other protocols based on IP, such as RIP, OSPF, and so on, refer to the configuration chapters for those protocols.

When parameter changes take effect

Most IP parameters described in this chapter are dynamic. They take effect immediately, as soon as you enter the CLI command. You can verify that a dynamic change has taken effect by displaying the running-config. To display the running-config, enter the **show running-config** or **write terminal** command at any CLI prompt.

To save a configuration change permanently so that the change remains in effect following a system reset or software reload, save the change to the startup-config file:

- To save configuration changes to the startup-config file, enter the **write memory** command from the Privileged EXEC level of any configuration level of the CLI.

Changes to memory allocation require you to reload the software after you save the changes to the startup-config file. When reloading the software is required to complete a configuration change described in this chapter, the procedure that describes the configuration change includes a step for reloading the software.

IP global parameters – Layer 3 Switches

Table 2 lists the IP global parameters for Layer 3 Switches.

TABLE 2 IP global parameters – Layer 3 Switches

Parameter	Description	Default	For more information
IP state	The Internet Protocol, version 4	Enabled **NOTE:** You cannot disable IP.	n/a
IP address and mask notation	Format for displaying an IP address and its network mask information. You can enable one of the following: • Class-based format; example: 192.168.1.1 255.255.255.0 • Classless Interdomain Routing (CIDR) format; example: 192.168.1.1/24	Class-based **NOTE:** Changing this parameter affects the display of IP addresses, but you can enter addresses in either format regardless of the display setting.	page 113
Router ID	The value that routers use to identify themselves to other routers when exchanging route information. OSPF and BGP4 use router IDs to identify routers. RIP does not use the router ID.	The IP address configured on the lowest-numbered loopback interface. If no loopback interface is configured, then the lowest-numbered IP address configured on the device.	page 31
Maximum Transmission Unit (MTU)	The maximum length an Ethernet packet can be without being fragmented.	1500 bytes for Ethernet II encapsulation 1492 bytes for SNAP encapsulation	page 28
Address Resolution Protocol (ARP)	A standard IP mechanism that routers use to learn the Media Access Control (MAC) address of a device on the network. The router sends the IP address of a device in the ARP request and receives the device MAC address in an ARP reply.	Enabled	page 35
ARP rate limiting	Lets you specify a maximum number of ARP packets the device will accept each second. If the device receives more ARP packets than you specify, the device drops additional ARP packets for the remainder of the one-second interval.	Disabled	page 36
ARP age	The amount of time the device keeps a MAC address learned through ARP in the device ARP cache. The device resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age. **NOTE:** You also can change the ARP age on an individual interface basis. Refer to Table 3 on page 15.	Ten minutes	page 37
Proxy ARP	An IP mechanism a router can use to answer an ARP request on behalf of a host, by replying with the router own MAC address instead of the host.	Disabled	page 38

TABLE 2 IP global parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	For more information
Static ARP entries	An ARP entry you place in the static ARP table. Static entries do not age out.	No entries	page 39
Time to Live (TTL)	The maximum number of routers (hops) through which a packet can pass before being discarded. Each router decreases a packet TTL by 1 before forwarding the packet. If decreasing the TTL causes the TTL to be 0, the router drops the packet instead of forwarding it.	64 hops	page 41
Directed broadcast forwarding	A directed broadcast is a packet containing all ones (or in some cases, all zeros) in the host portion of the destination IP address. When a router forwards such a broadcast, it sends a copy of the packet out each of its enabled IP interfaces. NOTE: You also can enable or disable this parameter on an individual interface basis. Refer to Table 3 on page 15.	Disabled	page 41
Directed broadcast mode	The packet format the router treats as a directed broadcast. The following formats can be directed broadcast: • All ones in the host portion of the packet destination address. • All zeroes in the host portion of the packet destination address.	All ones NOTE: If you enable all-zeroes directed broadcasts, all-ones directed broadcasts remain enabled.	page 42
Source-routed packet forwarding	A source-routed packet contains a list of IP addresses through which the packet must pass to reach its destination.	Enabled	page 41
Internet Control Message Protocol (ICMP) messages	The Brocade Layer 3 Switch can send the following types of ICMP messages: • Echo messages (ping messages) • Destination Unreachable messages	Enabled	page 43
ICMP Router Discovery Protocol (IRDP)	An IP protocol a router can use to advertise the IP addresses of its router interfaces to directly attached hosts. You can enable or disable the protocol, and change the following protocol parameters: • Forwarding method (broadcast or multicast) • Hold time • Maximum advertisement interval • Minimum advertisement interval • Router preference level NOTE: You also can enable or disable IRDP and configure the parameters on an individual interface basis. Refer to Table 3 on page 15.	Disabled	page 58
Reverse ARP (RARP)	An IP mechanism a host can use to request an IP address from a directly attached router when the host boots.	Enabled	page 61

TABLE 2 IP global parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	For more information
Static RARP entries	An IP address you place in the RARP table for RARP requests from hosts. NOTE: You must enter the RARP entries manually. The Layer 3 Switch does not have a mechanism for learning or dynamically generating RARP entries.	No entries	page 62
Maximum BootP relay hops	The maximum number of hops away a BootP server can be located from a router and still be used by the router clients for network booting.	Four	page 67
Domain name for Domain Name Server (DNS) resolver	A domain name (example: brocade.router.com) you can use in place of an IP address for certain operations such as IP pings, trace routes, and Telnet management connections to the router.	None configured	page 25
DNS default gateway addresses	A list of gateways attached to the router through which clients attached to the router can reach DNSs.	None configured	page 25
IP load sharing	A Brocade feature that enables the router to balance traffic to a specific destination across multiple equal-cost paths. IP load sharing uses a hashing algorithm based on the source IP address, destination IP address, protocol field in the IP header, TCP, and UDP information. NOTE: Load sharing is sometimes called Equal Cost Multi Path (ECMP).	Enabled	page 55
Maximum IP load sharing paths	The maximum number of equal-cost paths across which the Layer 3 Switch is allowed to distribute traffic.	Four	page 58
Origination of default routes	You can enable a router to originate default routes for the following route exchange protocols, on an individual protocol basis: • RIP • OSPF • BGP4	Disabled	page 144 page 178 page 291
Default network route	The router uses the default network route if the IP route table does not contain a route to the destination and also does not contain an explicit default route (0.0.0.0 0.0.0.0 or 0.0.0.0/0).	None configured	page 54

TABLE 2 IP global parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	For more information
Static route	An IP route you place in the IP route table.	No entries	page 45
Source interface	The IP address the router uses as the source address for Telnet, RADIUS, or TACACS/TACACS+ packets originated by the router. The router can select the source address based on either of the following: • The lowest-numbered IP address on the interface the packet is sent on. • The lowest-numbered IP address on a specific interface. The address is used as the source for all packets of the specified type regardless of interface the packet is sent on.	The lowest-numbered IP address on the interface the packet is sent on.	page 31

IP interface parameters – Layer 3 Switches

Table 3 lists the interface-level IP parameters for Layer 3 Switches.

TABLE 3 IP interface parameters – Layer 3 Switches

Parameter	Description	Default	For more information
IP state	The Internet Protocol, version 4	Enabled **NOTE:** You cannot disable IP.	n/a
IP address	A Layer 3 network interface address **NOTE:** Layer 2 Switches have a single IP address used for management access to the entire device. Layer 3 Switches have separate IP addresses on individual interfaces.	None configured[1]	page 19
Encapsulation type	The format of the packets in which the router encapsulates IP datagrams. The encapsulation format can be one of the following: • Ethernet II • SNAP	Ethernet II	page 28
Maximum Transmission Unit (MTU)	The maximum length (number of bytes) of an encapsulated IP datagram the router can forward.	1500 for Ethernet II encapsulated packets 1492 for SNAP encapsulated packets	page 30
ARP age	Locally overrides the global setting. Refer to Table 2 on page 12.	Ten minutes	page 37
Metric	A numeric cost the router adds to RIP routes learned on the interface. This parameter applies only to RIP routes.	1 (one)	page 144
Directed broadcast forwarding	Locally overrides the global setting. Refer to Table 2 on page 12.	Disabled	page 41
ICMP Router Discovery Protocol (IRDP)	Locally overrides the global IRDP settings. Refer to Table 2 on page 12.	Disabled	page 60

TABLE 3 IP interface parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	For more information
DHCP gateway stamp	The router can assist DHCP/BootP Discovery packets from one subnet to reach DHCP/BootP servers on a different subnet by placing the IP address of the router interface that receives the request in the request packet Gateway field. You can override the default and specify the IP address to use for the Gateway field in the packets. **NOTE:** UDP broadcast forwarding for client DHCP/BootP requests (bootps) must be enabled (this is enabled by default) and you must configure an IP helper address (the server IP address or a directed broadcast to the server subnet) on the port connected to the client.	The lowest-numbered IP address on the interface that receives the request	page 66
DHCP Client-Based Auto-Configuration	Allows the switch to obtain IP addresses from a DHCP host automatically, for either a specified (leased) or infinite period of time.	Enabled	page 80
DHCP Server	All FastIron devices can be configured to function as DHCP servers.	Disabled	page 67
UDP broadcast forwarding	The router can forward UDP broadcast packets for UDP applications such as BootP. By forwarding the UDP broadcasts, the router enables clients on one subnet to find servers attached to other subnets. **NOTE:** To completely enable a client UDP application request to find a server on another subnet, you must configure an IP helper address consisting of the server IP address or the directed broadcast address for the subnet that contains the server. See the next row.	The router helps forward broadcasts for the following UDP application protocols: • bootps • dns • netbios-dgm • netbios-ns • tacacs • tftp • time	page 63
IP helper address	The IP address of a UDP application server (such as a BootP or DHCP server) or a directed broadcast address. IP helper addresses allow the router to forward requests for certain UDP applications from a client on one subnet to a server on another subnet.	None configured	page 64

[1] Some devices have a factory default, used for troubleshooting during installation. For Layer 3 Switches, the address is on module 1 port 1 (or 1/1/1).

Basic IP parameters and defaults – Layer 2 Switches

IP is enabled by default. The following tables list the Layer 2 Switch IP parameters, their default values, and where to find configuration information.

NOTE
Brocade Layer 2 Switches also provide IP multicast forwarding, which is enabled by default. For more information about this feature, refer to the *Brocade ICX 6650 IP Multicast Configuration Guide*.

IP global parameters – Layer 2 Switches

Table 4 lists the IP global parameters for Layer 2 Switches.

TABLE 4 **IP global parameters – Layer 2 Switches**

Parameter	Description	Default	For more information
IP address and mask notation	Format for displaying an IP address and its network mask information. You can enable one of the following: • Class-based format; example: 192.168.1.1 255.255.255.0 • Classless Interdomain Routing (CIDR) format; example: 192.168.1.1/24	Class-based **NOTE:** Changing this parameter affects the display of IP addresses, but you can enter addresses in either format regardless of the display setting.	page 113
IP address	A Layer 3 network interface address **NOTE:** Layer 2 Switches have a single IP address used for management access to the entire device. Layer 3 Switches have separate IP addresses on individual interfaces.	None configured[1]	page 88
Default gateway	The IP address of a locally attached router (or a router attached to the Layer 2 Switch by bridges or other Layer 2 Switches). The Layer 2 Switch and clients attached to it use the default gateway to communicate with devices on other subnets.	None configured	page 88
Address Resolution Protocol (ARP)	A standard IP mechanism that networking devices use to learn the Media Access Control (MAC) address of another device on the network. The Layer 2 Switch sends the IP address of a device in the ARP request and receives the device MAC address in an ARP reply.	Enabled **NOTE:** You cannot disable ARP.	n/a
ARP age	The amount of time the device keeps a MAC address learned through ARP in the device ARP cache. The device resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age.	Ten minutes **NOTE:** You cannot change the ARP age on Layer 2 Switches.	n/a
Time to Live (TTL)	The maximum number of routers (hops) through which a packet can pass before being discarded. Each router decreases a packet TTL by 1 before forwarding the packet. If decreasing the TTL causes the TTL to be 0, the router drops the packet instead of forwarding it.	64 hops	page 90

TABLE 4 IP global parameters – Layer 2 Switches (Continued)

Parameter	Description	Default	For more information
Domain name for Domain Name Server (DNS) resolver	A domain name (example: brocade.router.com) you can use in place of an IP address for certain operations such as IP pings, trace routes, and Telnet management connections to the router.	None configured	page 89
DNS default gateway addresses	A list of gateways attached to the router through which clients attached to the router can reach DNSs.	None configured	page 89
Source interface	The IP address the Layer 2 Switch uses as the source address for Telnet, RADIUS, or TACACS/TACACS+ packets originated by the router. The Layer 2 Switch uses its management IP address as the source address for these packets.	The management IP address of the Layer 2 Switch. **NOTE:** This parameter is not configurable on Layer 2 Switches.	n/a
DHCP gateway stamp	The device can assist DHCP/BootP Discovery packets from one subnet to reach DHCP/BootP servers on a different subnet by placing the IP address of the router interface that forwards the packet in the packet Gateway field. You can specify up to 32 gateway lists. A gateway list contains up to eight gateway IP addresses. You activate DHCP assistance by associating a gateway list with a port. When you configure multiple IP addresses in a gateway list, the Layer 2 Switch inserts the addresses into the DHCP Discovery packets in a round robin fashion.	None configured	page 94
DHCP Client-Based Auto-Configuration	Allows the switch to obtain IP addresses from a DHCP host automatically, for either a specified (leased) or infinite period of time.	Enabled	page 80

[1] Some devices have a factory default, used for troubleshooting during installation. For Layer 3 Switches, the address is on port 1 (or 1/1/1).

Interface IP parameters – Layer 2 Switches

Table 5 lists the interface-level IP parameters for Layer 2 Switches.

TABLE 5 Interface IP parameters – Layer 2 Switches

Parameter	Description	Default	For more information
DHCP gateway stamp	You can configure a list of DHCP stamp addresses for a port. When the port receives a DHCP/BootP Discovery packet from a client, the port places the IP addresses in the gateway list into the packet Gateway field.	None configured	page 94

Configuring IP parameters – Layer 3 Switches

The following sections describe how to configure IP parameters. Some parameters can be configured globally while others can be configured on individual interfaces. Some parameters can be configured globally and overridden for individual interfaces.

NOTE
This section describes how to configure IP parameters for Layer 3 Switches. For IP configuration information for Layer 2 Switches, refer to "Configuring IP parameters – Layer 2 Switches" on page 88.

Configuring IP addresses

You can configure an IP address on the following types of Layer 3 Switch interfaces:

- Ethernet port
- Virtual routing interface (also called a Virtual Ethernet or "VE")
- Loopback interface

By default, you can configure up to 24 IP addresses on each interface.

You can increase this amount to up to 128 IP subnet addresses per port by increasing the size of the ip-subnet-port table.

Refer to the section "Displaying system parameter default values" in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

NOTE
Once you configure a virtual routing interface on a VLAN, you cannot configure Layer 3 interface parameters on individual ports. Instead, you must configure the parameters on the virtual routing interface itself.

Brocade devices support both classical IP network masks (Class A, B, and C subnet masks, and so on) and Classless Interdomain Routing (CIDR) network prefix masks:

- To enter a classical network mask, enter the mask in IP address format. For example, enter “192.168.22.99 255.255.255.0” for an IP address with a Class-C subnet mask.
- To enter a prefix network mask, enter a forward slash (/) and the number of bits in the mask immediately after the IP address. For example, enter “192.168.22.99/24” for an IP address that has a network mask with 24 significant bits (ones).

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the display to prefix format. Refer to “Changing the network mask display to prefix format” on page 113.

Assigning an IP address to an Ethernet port

To assign an IP address to port 1/1/1, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)# ip address 192.168.6.1 255.255.255.0
```

You also can enter the IP address and mask in CIDR format, as follows.

```
Brocade(config-if-e10000-1/1/1)# ip address 192.168.6.1/24
```

Syntax: [**no**] **ip address** *ip-addr ip-mask* [**ospf-ignore** | **ospf-passive** | **secondary**]

or

Syntax: [**no**] **ip address** *ip-addr/mask-bits* [**ospf-ignore** | **ospf-passive** | **secondary**]

The **ospf-ignore** | **ospf-passive** parameters modify the Layer 3 Switch defaults for adjacency formation and interface advertisement. Use one of these parameters if you are configuring multiple IP subnet addresses on the interface but you want to prevent OSPF from running on some of the subnets:

- **ospf-passive** – This option disables adjacency formation with OSPF neighbors. By default, when OSPF is enabled on an interface, the software forms OSPF router adjacencies between each primary IP address on the interface and the OSPF neighbor attached to the interface.
- **ospf-ignore** – This option disables OSPF adjacency formation and also disables advertisement of the interface into OSPF. The subnet is completely ignored by OSPF.

NOTE
The **ospf-passive** option disables adjacency formation but does not disable advertisement of the interface into OSPF. To disable advertisement in addition to disabling adjacency formation, you must use the **ospf-ignore** option.

Use the **secondary** parameter if you have already configured an IP address within the same subnet on the interface.

NOTE
When you configure more than one address in the same subnet, all but the first address are secondary addresses and do not form OSPF adjacencies.

NOTE
All physical IP interfaces on Brocade Layer 3 devices share the same MAC address. For this reason, if more than one connection is made between two devices, one of which is a Brocade Layer 3 device, Brocade recommends the use of virtual interfaces. It is not recommended to connect two or more physical IP interfaces between two routers.

Assigning an IP address to a loopback interface

Loopback interfaces are always up, regardless of the states of physical interfaces. They can add stability to the network because they are not subject to route flap problems that can occur due to unstable links between a Layer 3 Switch and other devices. You can configure up to eight loopback interfaces on a Chassis Layer 3 Switch .

You can add up to 24 IP addresses to each loopback interface.

NOTE
If you configure the Brocade Layer 3 Switch to use a loopback interface to communicate with a BGP4 neighbor, you also must configure a loopback interface on the neighbor and configure the neighbor to use that loopback interface to communicate with the Brocade Layer 3 Switch. Refer to "Adding a loopback interface" on page 292.

To add a loopback interface, enter commands such as those shown in the following example.

```
Brocade(config-bgp-router)# exit
Brocade(config)# interface loopback 1
Brocade(config-lbif-1)# ip address 10.0.0.1/24
```

Syntax: **interface loopback** *num*

The *num* parameter specifies the virtual interface number. You can specify from 1 to the maximum number of virtual interfaces supported on the device. To display the maximum number of virtual interfaces supported on the device, enter the **show default values** command. The maximum is listed in the System Parameters section, in the Current column of the virtual-interface row.

Refer to the syntax description in "Assigning an IP address to an Ethernet port" on page 20.

Assigning an IP address to a virtual interface

A virtual interface is a logical port associated with a Layer 3 Virtual LAN (VLAN) configured on a Layer 3 Switch. You can configure routing parameters on the virtual interface to enable the Layer 3 Switch to route protocol traffic from one Layer 3 VLAN to the other, without using an external router.[1]

You can configure IP routing interface parameters on a virtual interface. This section describes how to configure an IP address on a virtual interface. Other sections in this chapter that describe how to configure interface parameters also apply to virtual interfaces.

NOTE
The Layer 3 Switch uses the lowest MAC address on the device (the MAC address of port 1 or 1/1/1) as the MAC address for all ports within all virtual interfaces you configure on the device.

To add a virtual interface to a VLAN and configure an IP address on the interface, enter commands such as the following.

[1] The Brocade feature that allows routing between VLANs within the same device, without the need for external routers, is called Integrated Switch Routing (ISR).

```
Brocade(config)# vlan 2 name IP-Subnet_10.1.2.0/24
Brocade(config-vlan-2)# untag ethernet 1/1/1 to 1/1/4
Brocade(config-vlan-2)# router-interface ve1
Brocade(config-vlan-2)# interface ve1
Brocade(config-vif-1)# ip address 10.1.2.1/24
```

The first two commands in this example create a Layer 3 protocol-based VLAN name "IP-Subnet_10.1.2.0/24" and add a range of untagged ports to the VLAN. The **router-interface** command creates virtual interface 1 as the routing interface for the VLAN.

Syntax: **router-interface ve** *num*

The *num* variable specifies the virtual interface number. You can enter a number from 1 through 4095.

When configuring virtual routing interfaces on a device, you can specify a number from 1 through 4095. However, the total number of virtual routing interfaces that are configured must not exceed the system-max limit of 512. For more information on the number of virtual routing interfaces supported, refer to the section "Allocating memory for more VLANs or virtual routing interfaces" in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

The last two commands change to the interface configuration level for the virtual interface and assign an IP address to the interface.

Syntax: **interface ve** *num*

Refer to the syntax description in "Assigning an IP address to an Ethernet port" on page 20.

Configuring IP Follow on a virtual routing interface

IP Follow allows multiple virtual routing interfaces to share the same IP address. With this feature, one virtual routing interface is configured with an IP address, while the other virtual routing interfaces are configured to use that IP address, thus, they "follow" the virtual routing interface that has the IP address. This feature is helpful in conserving IP address space.

Configuration limitations and feature limitations for IP Follow on a virtual routing interface

- When configuring IP Follow, the primary virtual routing interface should not have ACL or DoS Protection configured. It is recommended that you create a dummy virtual routing interface as the primary and use the IP-follow virtual routing interface for the network.
- Global Policy Based Routing is not supported when IP Follow is configured.
- IPv6 is not supported with **ip-follow**.

Configuration syntax for IP Follow on a virtual routing interface

Configure IP Follow by entering commands such as the following.

```
Brocade(config)# vlan 2 name IP-Subnet_10.10.2.0/24
Brocade(config-vlan-2)# untag ethernet 1/1/1 to 1/1/4
Brocade(config-vlan-2)# router-interface ve1
Brocade(config-vlan-2)# interface ve 1
Brocade(config-vif-1)# ip address 10.10.2.1/24
Brocade(config-vif-1)# interface ve 2
Brocade(config-vif-2)# ip follow ve 1
Brocade(config-vif-2)# interface ve 3
Brocade(config-vif-3)# ip follow ve 1
```

Syntax: [**no**] **ip follow ve** *number*

For *number,* enter the ID of the virtual routing interface.

Use the **no** form of the command to disable the configuration.

Virtual routing interface 2 and 3 do not have their own IP subnet addresses, but are sharing the IP address of virtual routing interface 1.

Deleting an IP address

To delete an IP address, enter the **no ip address** command.

```
Brocade(config-if-e10000-1/1/1)# no ip address 10.1.2.1
```

This command deletes IP address 10.1.2.1. You do not need to enter the subnet mask.

To delete all IP addresses from an interface, enter the **no ip address** * command.

```
Brocade(config-if-e10000-1/1/1)# no ip address *
```

Syntax: no ip address *ip-addr* | *

Configuring 31-bit subnet masks on point-to-point networks

To conserve IPv4 address space, a 31-bit subnet mask can be assigned to point-to-point networks. Support for an IPv4 address with a 31-bit subnet mask is described in RFC 3021.

With IPv4, four IP addresses with a 30-bit subnet mask are allocated on point-to-point networks. In contrast, a 31-bit subnet mask uses only two IP addresses: all zero bits and all one bits in the host portion of the IP address. The two IP addresses are interpreted as host addresses, and do not require broadcast support because any packet that is transmitted by one host is always received by the other host at the receiving end. Therefore, directed broadcast on a point-to-point interface is eliminated.

IP-directed broadcast CLI configuration at the global level, or the per interface level, is not applicable on interfaces configured with a 31-bit subnet mask IP address.

When the 31-bit subnet mask address is configured on a point-to-point link, using network addresses for broadcast purposes is not allowed. For example, in an IPV4 broadcast scheme, the following subnets can be configured:

- 10.10.10.1 - Subnet for directed broadcast: {<Network-number>, -1}
- 10.10.10.0 - Subnet for network address: {<Network-number>, 0}

In a point-to-point link with a 31-bit subnet mask, the previous two addresses are interpreted as host addresses and packets are not rebroadcast.

Configuring an IPv4 address with a 31-bit subnet mask

To configure an IPv4 address with a 31-bit subnet mask, enter the following commands.

You can configure an IPv4 address with a 31-bit subnet mask on any interface (for example, Ethernet, loopback, VE, or tunnel interfaces).

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip address 10.10.9.9 255.255.255.254
```

You can also enter the IP address and mask in the Classless Inter-domain Routing (CIDR) format, as follows.

```
Brocade(config-if-e10000-1/1/5)# ip address 10.10.9.9/31
```

Syntax: [**no**] **ip address** *ip-address ip-mask*

Syntax: [**no**] **ip address** *ip-address/subnet mask-bits*

The *ip-address* variable specifies the host address. The *ip-mask* variable specifies the IP network mask. The *subnet mask-bits* variable specifies the network prefix mask.

To disable configuration for an IPv4 address with a 31-bit subnet mask on any interface, use the **no** form of the command.

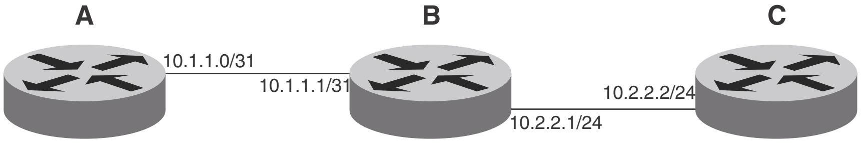
You cannot configure a secondary IPv4 address with a 31-bit subnet mask on any interface. The following error message is displayed when a secondary IPv4 address with a 31-bit subnet mask is configured.

```
Error: Cannot assign /31 subnet address as secondary
```

Configuration example

Figure 2 shows the usage of 31- and 24-bit subnet masks in configuring IP addresses.

FIGURE 2 **Configured 31- bit and 24-bit subnet masks**



Router A is connected to Router B as a point-to-point link with 10.1.1.0/31 subnet. There are only two available addresses in this subnet, 10.1.1.0 on Router A and 10.1.1.1 on Router B,

Routers B and C are connected by a regular 24-bit subnet. Router C can either be a switch with many hosts belonging to the 10.2.2.2/24 subnet connected to it, or it can be a router.

Router A

```
RouterA(config)# interface ethernet 1/1/1
RouterA(config-if-e10000-1/1/1)# ip address 10.1.1.0/31
```

Router B

```
RouterB(config)# interface ethernet 1/1/1
RouterB(config-if-e10000-1/1/1)# ip address 10.1.1.1/31
RouterB(config-if-e10000-1/1/1)# exit
RouterB(config# interface ethernet 1/3/1
RouterB(config-if-e10000-1/3/1)# ip address 10.2.2.1/24
```

Router C

```
RouterC(config# interface ethernet 1/3/1
RouterC(config-if-e10000-1/3/1)# ip address 10.2.2.2/24
```

Displaying information for a 31-bit subnet mask

Use the following commands to display information for the 31-bit subnet mask:

- show run interface
- show ip route
- show ip cache

Configuring DNS resolver

The Domain Name System (DNS) resolver is a feature in a Layer 2 or Layer 3 switch that sends and receives queries to and from the DNS server on behalf of a client.

You can create a list of domain names that can be used to resolve host names. This list can have more than one domain name. When a client performs a DNS query, all hosts within the domains in the list can be recognized and queries can be sent to any domain on the list.

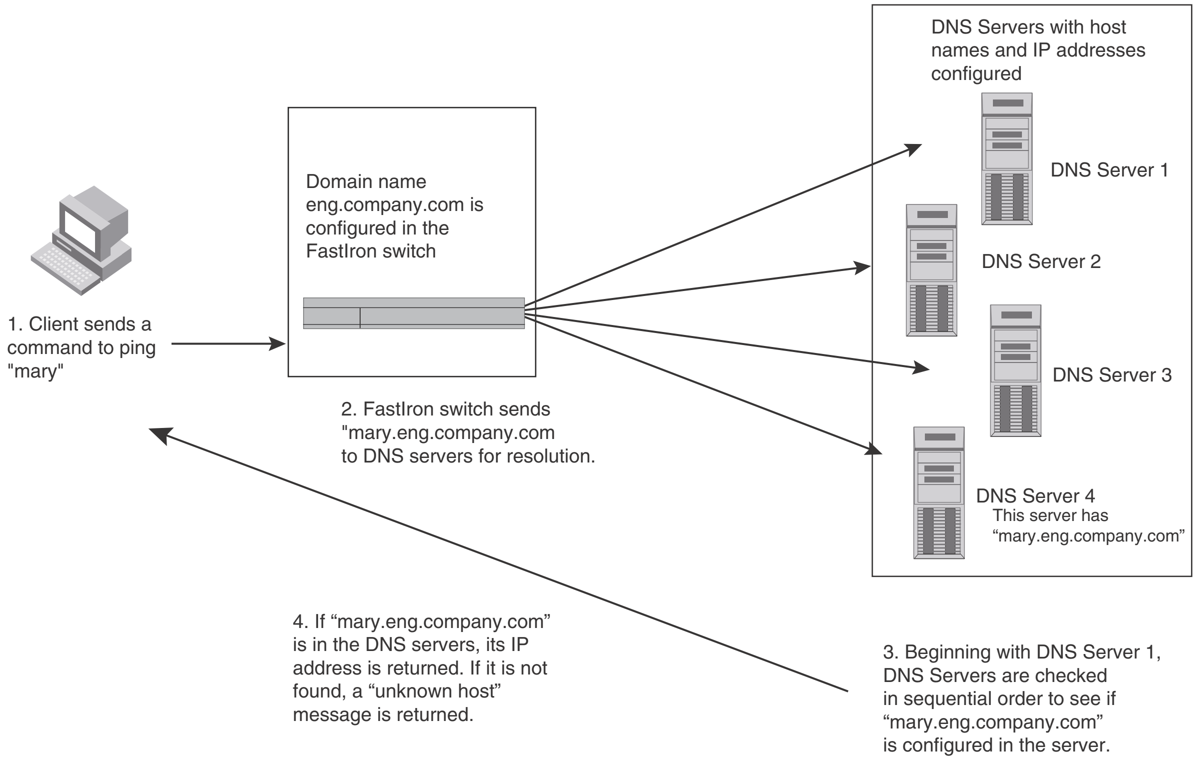
After you define a domain name, the Brocade device automatically appends the appropriate domain to a host and forwards it to the DNS servers for resolution.

For example, if the domain “ds.company.com” is defined on a Layer 2 or Layer 3 switch and you want to initiate a ping to “mary”, you must reference only the host name instead of the host name and its domain name. For example, you could enter the following command to initiate the ping.

```
U:> ping mary
```

The Layer 2 or Layer 3 switch qualifies the host name by appending a domain name (for example, mary.ds1.company.com). This qualified name is sent to the DNS server for resolution. If there are four DNS servers configured, it is sent to the first DNS server. If the host name is not resolved, it is sent to the second DNS server. If a match is found, a response is sent back to the client with the host IP address. If no match is found, an “unknown host” message is returned. (Refer to Figure 3.)

FIGURE 3 DNS resolution with one domain name


DNS Servers with host names and IP addresses configured
Domain name eng.company.com is configured in the FastIron switch
DNS Server 1
DNS Server 2
DNS Server 3
DNS Server 4
This server has "mary.eng.company.com"
1. Client sends a command to ping "mary"
2. FastIron switch sends "mary.eng.company.com to DNS servers for resolution.
4. If "mary.eng.company.com" is in the DNS servers, its IP address is returned. If it is not found, a "unknown host" message is returned.
3. Beginning with DNS Server 1, DNS Servers are checked in sequential order to see if "mary.eng.company.com" is configured in the server.

Defining a domain name

To define a domain to resolve host names, enter the **ip dns domain-name** command.

```
Brocade(config)# ip dns domain-name ds.company.com
```

Syntax: [**no**] **ip dns domain-name** *domain-name*

Enter the domain name for *domain-name.*

Defining DNS server addresses

You can configure the Brocade device to recognize up to four DNS servers. The first entry serves as the primary default address. If a query to the primary address fails to be resolved after three attempts, the next DNS address is queried (also up to three times). This process continues for each defined DNS address until the query is resolved. The order in which the default DNS addresses are polled is the same as the order in which you enter them.

To define DNS servers, enter the **ip dns server-address** command.

```
Brocade(config)# ip dns server-address 192.168.22.199 192.168.7.15 192.168.10.25
192.168.20.15
```

Syntax: [**no**] **ip dns server-address** *ip-addr* [*ip-addr*] [*ip-addr*] [*ip-addr*]

In this example, the first IP address entered becomes the primary DNS address and all others are secondary addresses. Because IP address 192.168.20.15 is the last address listed, it is also the last address consulted to resolve a query.

Defining a domain list

If you want to use more than one domain name to resolve host names, you can create a list of domain names. For example, enter the commands such as the following.

```
Brocade(config)# ip dns domain-list company.com
Brocade(config)# ip dns domain-list ds.company.com
Brocade(config)# ip dns domain-list hw_company.com
Brocade(config)# ip dns domain-list qa_company.com
Brocade(config)#
```

The domain names are tried in the order you enter them

Syntax: [**no**] **ip dns domain-list** *domain-name*

Using a DNS name to initiate a trace route

Suppose you want to trace the route from a Brocade Layer 3 Switch to a remote server identified as NYC02 on domain newyork.com. Because the NYC02@ds1.newyork.com domain is already defined on the Layer 3 Switch, you need to enter only the host name, NYC02, as noted in the following example.

```
Brocade# traceroute nyc02
```

Syntax: **traceroute** *host-ip-addr* [**maxttl** *value*] [**minttl** *value*] [**numeric**] [**timeout** *value*]
[**source-ip** *ip addr*]

The only required parameter is the IP address of the host at the other end of the route.

After you enter the command, a message indicating that the DNS query is in process and the current gateway address (IP address of the domain name server) being queried appear on the screen.

```
Type Control-c to abort
Sending DNS Query to 192.168.22.199
Tracing Route to IP node 192.168.22.80
To ABORT Trace Route, Please use stop-traceroute command.
 Traced route to target IP node 192.168.22.80:
   IP Address         Round Trip Time1    Round Trip Time2
  192.168.6.30        93 msec             121 msec
```

NOTE
In the previousexample, 192.168.22.199 is the IP address of the domain name server (default DNS gateway address), and 192.168.22.80 represents the IP address of the NYC02 host.

Configuring packet parameters

You can configure the following packet parameters on Layer 3 Switches. These parameters control how the Layer 3 Switch sends IP packets to other devices on an Ethernet network. The Layer 3 Switch always places IP packets into Ethernet packets to forward them on an Ethernet port.

- **Encapsulation type** – The format for the Layer 2 packets within which the Layer 3 Switch sends IP packets.
- **Maximum Transmission Unit (MTU)** – The maximum length of IP packet that a Layer 2 packet can contain. IP packets that are longer than the MTU are fragmented and sent in multiple Layer 2 packets. You can change the MTU globally or an individual ports:
  - **Global MTU** – The default MTU value depends on the encapsulation type on a port and is 1500 bytes for Ethernet II encapsulation and 1492 bytes for SNAP encapsulation.
  - **Port MTU** – A port default MTU depends on the encapsulation type enabled on the port.

Changing the encapsulation type

The Layer 3 Switch encapsulates IP packets into Layer 2 packets, to send the IP packets on the network. (A Layer 2 packet is also called a MAC layer packet or an Ethernet frame.) The source address of a Layer 2 packet is the MAC address of the Layer 3 Switch interface sending the packet. The destination address can be one of the following:

- The MAC address of the IP packet destination. In this case, the destination device is directly connected to the Layer 3 Switch.
- The MAC address of the next-hop gateway toward the packet destination.
- An Ethernet broadcast address.

The entire IP packet, including the source and destination address and other control information and the data, is placed in the data portion of the Layer 2 packet. Typically, an Ethernet network uses one of two different formats of Layer 2 packet:

- Ethernet II
- Ethernet SNAP (also called IEEE 802.3)

The control portions of these packets differ slightly. All IP devices on an Ethernet network must use the same format. Brocade Layer 3 Switches use Ethernet II by default. You can change the IP encapsulation to Ethernet SNAP on individual ports if needed.

NOTE
All devices connected to the Layer 3 Switch port must use the same encapsulation type.

To change the IP encapsulation type on interface 5 to Ethernet SNAP, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip encapsulation snap
```

Syntax: ip encapsulation snap | ethernet_ii

Changing the MTU

The Maximum Transmission Unit (MTU) is the maximum length of IP packet that a Layer 2 packet can contain. IP packets that are longer than the MTU are fragmented and sent in multiple Layer 2 packets. You can change the MTU globally or on individual ports.

The default MTU is 1500 bytes for Ethernet II packets and 1492 for Ethernet SNAP packets.

MTU enhancements

Brocade devices contain the following enhancements to jumbo packet support:

- Hardware forwarding of Layer 3 jumbo packets – Layer 3 IP unicast jumbo packets received on a port that supports the frame MTU size and forwarded to another port that also supports the frame MTU size are forwarded in hardware. .
- ICMP unreachable message if a frame is too large to be forwarded – If a jumbo packet has the Do not Fragment (DF) bit set, and the outbound interface does not support the packet MTU size, the Brocade device sends an ICMP unreachable message to the device that sent the packet.

NOTE
These enhancements apply only to transit traffic forwarded through the Brocade device.

Configuration considerations for increasing the MTU

- The MTU command is applicable to VEs and physical IP interfaces. It applies to traffic routed between networks.
- You cannot use this command to set Layer 2 maximum frame sizes per interface. The global **jumbo** command causes all interfaces to accept Layer 2 frames.
- When you increase the MTU size of a port, the increase uses system resources. Increase the MTU size only on the ports that need it. For example, if you have one port connected to a server that uses jumbo frames and two other ports connected to clients that can support the jumbo frames, increase the MTU only on those three ports. Leave the MTU size on the other ports at the default value (1500 bytes). Globally increase the MTU size only if needed.

Forwarding traffic to a port with a smaller MTU size

In order to forward traffic from a port with 1500 MTU configured to a port that has a smaller MTU (for example, 750) size, you must apply the **mtu-exceed forward** global command. To remove this setting, enter the **mtu-exceed hard-drop** command. **MTU-exceed hard-drop** is the default state of the router.

Syntax:**mtu-exceed** [**forward** | **hard-drop**]

- **forward** - forwards a packet from a port with a larger MTU to a port with a smaller MTU
- **hard-drop** - resets to default, removes the forward function.

Globally changing the Maximum Transmission Unit

The Maximum Transmission Unit (MTU) is the maximum size an IP packet can be when encapsulated in a Layer 2 packet. If an IP packet is larger than the MTU allowed by the Layer 2 packet, the Layer 3 Switch fragments the IP packet into multiple parts that will fit into the Layer 2 packets, and sends the parts of the fragmented IP packet separately, in different Layer 2 packets. The device that receives the multiple fragments of the IP packet reassembles the fragments into the original packet.

You can increase the MTU size to accommodate jumbo packet sizes up to 10,240 bytes.

To globally enable jumbo support on all ports of a Brocade ICX 6650 device, enter commands such as the following.

```
Brocade(config)# jumbo
Brocade(config)# write memory
Brocade(config)# end
Brocade# reload
```

Syntax: [**no**] **jumbo**

NOTE
You must save the configuration change and then reload the software to enable jumbo support.

Changing the MTU on an individual port

By default, the maximum Ethernet MTU sizes are as follows:

- 1500 bytes – The maximum for Ethernet II encapsulation
- 1492 bytes – The maximum for SNAP encapsulation

When jumbo mode is enabled, the maximum Ethernet MTU sizes are as follows:

- 10,240 bytes– The maximum for Ethernet II encapsulation
- 10,240 bytes – The maximum for SNAP encapsulation

NOTE
If you set the MTU of a port to a value lower than the global MTU and from 576 through 1499, the port fragments the packets. However, if the port MTU is exactly 1500 and this is larger than the global MTU, the port drops the packets.

NOTE
You must save the configuration change and then reload the software to enable jumbo support.

To change the MTU for interface 1/1/5 to 1000, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip mtu 1000
Brocade(config-if-e10000-1/1/5)# write memory
Brocade(config-if-e10000-1/1/5)# end
Brocade# reload
```

Syntax: [**no**] **ip mtu** *num*

The *num* parameter specifies the MTU. Ethernet II packets can hold IP packets from 576 through 1500 bytes long. If jumbo mode is enabled, Ethernet II packets can hold IP packets up to 10,240 bytes long. Ethernet SNAP packets can hold IP packets from 576 through 1492 bytes long. If jumbo mode is enabled, SNAP packets can hold IP packets up to 10,240 bytes long. The default MTU for Ethernet II packets is 1500. The default MTU for SNAP packets is 1492.

Path MTU discovery (RFC 1191) support

Brocade ICX 6650 devices support the path MTU discovery method described in RFC 1191. When the Brocade device receives an IP packet that has its Do not Fragment (DF) bit set, and the packet size is greater than the MTU value of the outbound interface, then the Brocade device returns an ICMP Destination Unreachable message to the source of the packet, with the Code indicating "fragmentation needed and DF set". The ICMP Destination Unreachable message includes the MTU of the outbound interface. The source host can use this information to help determine the maximum MTU of a path to a destination.

RFC 1191 is supported on all interfaces.

Changing the router ID

In most configurations, a Layer 3 Switch has multiple IP addresses, usually configured on different interfaces. As a result, a Layer 3 Switch identity to other devices varies depending on the interface to which the other device is attached. Some routing protocols, including Open Shortest Path First (OSPF) and Border Gateway Protocol version 4 (BGP4), identify a Layer 3 Switch by just one of the IP addresses configured on the Layer 3 Switch, regardless of the interfaces that connect the Layer 3 Switches. This IP address is the router ID.

NOTE
Routing Information Protocol (RIP) does not use the router ID.

NOTE
If you change the router ID, all current BGP4 sessions are cleared.

By default, the router ID on a Brocade Layer 3 Switch is one of the following:

- If the router has loopback interfaces, the default router ID is the IP address configured on the lowest numbered loopback interface configured on the Layer 3 Switch. For example, if you configure loopback interfaces 1, 2, and 3 as follows, the default router ID is 192.168.9.9/24:
  - Loopback interface 1, 192.168.9.9/24
  - Loopback interface 2, 192.168.4.4/24
  - Loopback interface 3, 192.168.1.1/24
- If the device does not have any loopback interfaces, the default router ID is the lowest numbered IP interface configured on the device.

If you prefer, you can explicitly set the router ID to any valid IP address. The IP address cannot be in use on another device in the network.

NOTE
Brocade Layer 3 Switches use the same router ID for both OSPF and BGP4. If the router is already configured for OSPF, you may want to use the router ID that is already in use on the router rather than set a new one. To display the router ID, enter the **show ip** command at any CLI level.

To change the router ID, enter a command such as the following.

```
Brocade(config)# ip router-id 192.168.22.26
```

Syntax: **ip router-id** *ip-addr*

The *ip-addr* can be any valid, unique IP address.

NOTE
You can specify an IP address used for an interface on the Brocade Layer 3 Switch, but do not specify an IP address in use by another device.

Specifying a single source interface for specified packet types

When the Layer 3 Switch originates a packet of one of the following types, the source address of the packet is the lowest-numbered IP address on the interface that sends the packet:

- Telnet

- TACACS/TACACS+
- TFTP
- RADIUS
- Syslog
- SNTP
- SSH
- SNMP traps

You can configure the Layer 3 Switch to always use the lowest-numbered IP address on a specific Ethernet, loopback, or virtual interface as the source addresses for these packets. When configured, the Layer 3 Switch uses the same IP address as the source for all packets of the specified type, regardless of the ports that actually sends the packets.

Identifying a single source IP address for specified packets provides the following benefits:

- If your server is configured to accept packets only from specific IP addresses, you can use this feature to simplify configuration of the server by configuring the Brocade device to always send the packets from the same link or source address.
- If you specify a loopback interface as the single source for specified packets, servers can receive the packets regardless of the states of individual links. Thus, if a link to the server becomes unavailable but the client or server can be reached through another link, the client or server still receives the packets, and the packets still have the source IP address of the loopback interface.

The software contains separate CLI commands for specifying the source interface for specific packets. You can configure a source interface for one or more of these types of packets separately.

The following sections show the syntax for specifying a single source IP address for specific packet types.

Telnet packets

To specify the IP address configured on a virtual interface as the device source for all Telnet packets, enter commands such as the following.

```
Brocade(config)# interface loopback 2
Brocade(config-lbif-2)# ip address 10.0.0.2/24
Brocade(config-lbif-2)# exit
Brocade(config)# ip telnet source-interface loopback 2
```

The commands in this example configure loopback interface 2, assign IP address 10.0.0.2/24 to the interface, then designate the interface as the source for all Telnet packets from the Layer 3 Switch.

The following commands configure an IP interface on an Ethernet port and designate the address port as the source for all Telnet packets from the Layer 3 Switch.

```
Brocade(config)# interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)# ip address 192.168.22.110/24
Brocade(config-if-e10000-1/1/4)# exit
Brocade(config)# ip telnet source-interface ethernet 1/1/4
```

Syntax: [**no**] **ip telnet source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

TACACS/TACACS+ packets

To specify the lowest-numbered IP address configured on a virtual interface as the device source for all TACACS/TACACS+ packets, enter commands such as the following.

```
Brocade(config)# interface ve 1
Brocade(config-vif-1)# ip address 10.0.0.3/24
Brocade(config-vif-1)# exit
Brocade(config)# ip tacacs source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface as the source for all TACACS/TACACS+ packets from the Layer 3 Switch.

Syntax: [**no**] **ip tacacs source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

RADIUS packets

To specify the lowest-numbered IP address configured on a virtual interface as the device source for all RADIUS packets, enter commands such as the following.

```
Brocade(config)# interface ve 1
Brocade(config-vif-1)# ip address 10.0.0.3/24
Brocade(config-vif-1)# exit
Brocade(config)# ip radius source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface as the source for all RADIUS packets from the Layer 3 Switch.

Syntax: [**no**] **ip radius source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

TFTP packets

To specify the lowest-numbered IP address configured on a virtual interface as the device source for all TFTP packets, enter commands such as the following.

```
Brocade(config)# interface ve 1
Brocade(config-vif-1)# ip address 10.0.0.3/24
Brocade(config-vif-1)# exit
Brocade(config)# ip tftp source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface's address as the source address for all TFTP packets.

Syntax: [**no**] **ip tftp source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

The default is the lowest-numbered IP address configured on the port through which the packet is sent. The address therefore changes, by default, depending on the port.

Syslog packets

To specify the lowest-numbered IP address configured on a virtual interface as the device source for all Syslog packets, enter commands such as the following.

```
Brocade(config)# interface ve 1
Brocade(config-vif-1)# ip address 10.0.0.4/24
Brocade(config-vif-1)# exit
Brocade(config)# ip syslog source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.4/24 to the interface, then designate the interface's address as the source address for all Syslog packets.

Syntax: [**no**] **ip syslog source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

The default is the lowest-numbered IP or IPv6 address configured on the port through which the packet is sent. The address therefore changes, by default, depending on the port.

SNTP packets

To specify the lowest-numbered IP address configured on a virtual interface as the device source for all SNTP packets, enter commands such as the following.

```
Brocade(config)# interface ve 1
Brocade(config-vif-1)# ip address 10.0.0.5/24
Brocade(config-vif-1)# exit
Brocade(config)# ip sntp source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.5/24 to the interface, then designate the interface's address as the source address for all SNTP packets.

Syntax: [no] **ip sntp source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

The default is the lowest-numbered IP or IPv6 address configured on the port through which the packet is sent. The address therefore changes, by default, depending on the port.

SSH packets

NOTE
When you specify a single SSH source, you can use only that source address to establish SSH management sessions with the Brocade device.

To specify the numerically lowest IP address configured on a loopback interface as the device source for all SSH packets, enter commands such as a the following.

```
Brocade(config)# interface loopback 2
Brocade(config-lbif-2)# ip address 10.0.0.2/24
Brocade(config-lbif-2)# exit
Brocade(config)# ip ssh source-interface loopback 2
```

The commands in this example configure loopback interface 2, assign IP address 10.0.0.2/24 to the interface, then designate the interface as the source for all SSH packets from the Layer 3 Switch.

Syntax: [**no**] **ip ssh source-interface ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num* | management *num*

The *num* variable is a loopback interface, virtual interface or management interface number.

SNMP packets

To specify a loopback interface as the SNMP single source trap, enter commands such as the following.

```
Brocade(config)# interface loopback 1
Brocade(config-lbif-1)# ip address 10.0.0.1/24
Brocade(config-lbif-1)# exit
Brocade(config)# snmp-server trap-source loopback 1
```

The commands in this example configure loopback interface 1, assign IP address 10.0.0.1/24 to the loopback interface, then designate the interface as the SNMP trap source for this device. Regardless of the port the Brocade device uses to send traps to the receiver, the traps always arrive from the same source IP address.

Syntax: [**no**] **snmp-server trap-source ethernet** *stack-unit/slotnum/portnum* | **loopback** *num* | **ve** *num*

The *num* variable is a loopback interface or virtual interface number.

ARP parameter configuration

Address Resolution Protocol (ARP) is a standard IP protocol that enables an IP Layer 3 Switch to obtain the MAC address of another device interface when the Layer 3 Switch knows the IP address of the interface. ARP is enabled by default and cannot be disabled.

NOTE
Brocade Layer 2 Switches also support ARP. The description in "How ARP works" also applies to ARP on Brocade Layer 2 Switches. However, the configuration options described later in this section apply only to Layer 3 Switches, not to Layer 2 Switches.

How ARP works

A Layer 3 Switch needs to know a destination MAC address when forwarding traffic, because the Layer 3 Switch encapsulates the IP packet in a Layer 2 packet (MAC layer packet) and sends the Layer 2 packet to a MAC interface on a device directly attached to the Layer 3 Switch. The device can be the packet final destination or the next-hop router toward the destination.

The Layer 3 Switch encapsulates IP packets in Layer 2 packets regardless of whether the ultimate destination is locally attached or is multiple router hops away. Since the Layer 3 Switch IP route table and IP forwarding cache contain IP address information but not MAC address information, the Layer 3 Switch cannot forward IP packets based solely on the information in the route table or forwarding cache. The Layer 3 Switch needs to know the MAC address that corresponds with the IP address of either the packet locally attached destination or the next-hop router that leads to the destination.

For example, to forward a packet whose destination is multiple router hops away, the Layer 3 Switch must send the packet to the next-hop router toward its destination, or to a default route or default network route if the IP route table does not contain a route to the packet destination. In each case, the Layer 3 Switch must encapsulate the packet and address it to the MAC address of a locally attached device, the next-hop router toward the IP packet destination.

To obtain the MAC address required for forwarding a datagram, the Layer 3 Switch does the following:

- First, the Layer 3 Switch looks in the ARP cache (not the static ARP table) for an entry that lists the MAC address for the IP address. The ARP cache maps IP addresses to MAC addresses. The cache also lists the port attached to the device and, if the entry is dynamic, the age of the entry. A dynamic ARP entry enters the cache when the Layer 3 Switch receives an ARP reply or receives an ARP request (which contains the sender IP address and MAC address). A static entry enters the ARP cache from the static ARP table (which is a separate table) when the interface for the entry comes up.

  To ensure the accuracy of the ARP cache, each dynamic entry has its own age timer. The timer is reset to zero each time the Layer 3 Switch receives an ARP reply or ARP request containing the IP address and MAC address of the entry. If a dynamic entry reaches its maximum allowable age, the entry times out and the software removes the entry from the table. Static entries do not age out and can be removed only by you.

- If the ARP cache does not contain an entry for the destination IP address, the Layer 3 Switch broadcasts an ARP request out all its IP interfaces. The ARP request contains the IP address of the destination. If the device with the IP address is directly attached to the Layer 3 Switch, the device sends an ARP response containing its MAC address. The response is a unicast packet addressed directly to the Layer 3 Switch. The Layer 3 Switch places the information from the ARP response into the ARP cache.

  ARP requests contain the IP address and MAC address of the sender, so all devices that receive the request learn the MAC address and IP address of the sender and can update their own ARP caches accordingly.

  NOTE
  The ARP request broadcast is a MAC broadcast, which means the broadcast goes only to devices that are directly attached to the Layer 3 Switch. A MAC broadcast is not routed to other networks. However, some routers, including Brocade Layer 3 Switches, can be configured to reply to ARP requests from one network on behalf of devices on another network. Refer to "Enabling proxy ARP" on page 38.

NOTE
If the router receives an ARP request packet that it is unable to deliver to the final destination because of the ARP timeout and no ARP response is received (the Layer 3 Switch knows of no route to the destination address), the router sends an ICMP Host Unreachable message to the source.

Rate limiting ARP packets

You can limit the number of ARP packets the Brocade device accepts during each second. By default, the software does not limit the number of ARP packets the device can receive. Since the device sends ARP packets to the CPU for processing, if a device in a busy network receives a high number of ARP packets in a short period of time, some CPU processing might be deferred while the CPU processes the ARP packets.

To prevent the CPU from becoming flooded by ARP packets in a busy network, you can restrict the number of ARP packets the device will accept each second. When you configure an ARP rate limit, the device accepts up to the maximum number of packets you specify, but drops additional ARP packets received during the one-second interval. When a new one-second interval starts, the counter restarts at zero, so the device again accepts up to the maximum number of ARP packets you specified, but drops additional packets received within the interval.

To limit the number of ARP packets the device will accept each second, enter the **rate-limit-arp** command at the global CONFIG level of the CLI.

```
Brocade(config)# rate-limit-arp 100
```

This command configures the device to accept up to 100 ARP packets each second. If the device receives more than 100 ARP packets during a one-second interval, the device drops the additional ARP packets during the remainder of that one-second interval.

Syntax: [**no**] **rate-limit-arp** *num*

The *num* parameter specifies the number of ARP packets and can be from 0 through 100. If you specify 0, the device will not accept any ARP packets.

NOTE
If you want to change a previously configured the ARP rate limiting policy, you must remove the previously configured policy using the **no rate-limit-arp** *num* command before entering the new policy.

Changing the ARP aging period

When the Layer 3 Switch places an entry in the ARP cache, the Layer 3 Switch also starts an aging timer for the entry. The aging timer ensures that the ARP cache does not retain learned entries that are no longer valid. An entry can become invalid when the device with the MAC address of the entry is no longer on the network.

The ARP age affects dynamic (learned) entries only, not static entries. The default ARP age is ten minutes. On Layer 3 Switches, you can change the ARP age to a value from 0 through 240 minutes. You cannot change the ARP age on Layer 2 Switches. If you set the ARP age to zero, aging is disabled and entries do not age out.

To globally change the ARP aging parameter to 20 minutes, enter the **ip arp-age** command.

```
Brocade(config)# ip arp-age 20
```

Syntax: **ip arp-age** *num*

The *num* parameter specifies the number of minutes and can be from 0 through 240. The default is 10. If you specify 0, aging is disabled.

To override the globally configured IP ARP age on an individual interface, enter a command such as the following at the interface configuration level.

```
Brocade(config-if-e10000-1/1/1)# ip arp-age 30
```

Syntax: [**no**] **ip arp-age** *num*

The *num* parameter specifies the number of minutes and can be from 0 through 240. The default is the globally configured value, which is 10 minutes by default. If you specify 0, aging is disabled.

Enabling proxy ARP

Proxy ARP allows a Layer 3 Switch to answer ARP requests from devices on one network on behalf of devices in another network. Since ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Thus, ARP requests do not cross routers.

For example, if Proxy ARP is enabled on a Layer 3 Switch connected to two subnets, 192.168.10.0/24 and 192.168.20.0/24, the Layer 3 Switch can respond to an ARP request from 192.168.10.69 for the MAC address of the device with IP address 192.168.20.69. In standard ARP, a request from a device in the 192.168.10.0/24 subnet cannot reach a device in the 192.168.20.0 subnet if the subnets are on different network cables, and thus is not answered.

NOTE
An ARP request from one subnet can reach another subnet when both subnets are on the same physical segment (Ethernet cable), because MAC-layer broadcasts reach all the devices on the segment.

Proxy ARP is disabled by default on Brocade Layer 3 Switches. This feature is not supported on Brocade Layer 2 Switches.

You can enable proxy ARP at the Interface level, as well as at the Global CONFIG level, of the CLI.

NOTE
Configuring proxy ARP at the Interface level overrides the global configuration.

Enabling proxy ARP globally

To enable IP proxy ARP on a global basis, enter the **ip proxy-arp** command.

```
Brocade(config)# ip proxy-arp
```

To again disable IP proxy ARP on a global basis, enter the **no ip proxy-arp** command.

```
Brocade(config)# no ip proxy-arp
```

Syntax: [**no**] **ip proxy-arp**

Enabling IP ARP on an interface

NOTE
Configuring proxy ARP at the Interface level overrides the global configuration.

To enable IP proxy ARP on an interface, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip proxy-arp enable
```

To again disable IP proxy ARP on an interface, enter the following command.

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip proxy-arp disable
```

Syntax: [**no**] **ip proxy-arp enable** | **disable**

Enabling local proxy ARP

Brocade devices support Proxy Address Resolution Protocol (*Proxy ARP*), a feature that enables router ports to respond to ARP requests for subnets it can reach. However, router ports will not respond to ARP requests for IP addresses in the same subnet as the incoming ports, unless Local Proxy ARP per IP interface is enabled. *Local Proxy ARP* enables router ports to reply to ARP requests for IP addresses within the same subnet and to forward all traffic between hosts in the subnet.

When Local Proxy ARP is enabled on a router port, the port will respond to ARP requests for IP addresses within the same subnet, if it has ARP entries for the destination IP addresses in the ARP cache. If it does not have ARP entries for the IP addresses, the port will attempt to resolve them by broadcasting its own ARP requests.

Local Proxy ARP is disabled by default. To use Local Proxy ARP, Proxy ARP (**ip proxy-arp** command) must be enabled globally on the Brocade device. You can enter the CLI command to enable Local Proxy ARP even though Proxy ARP is not enabled, however, the configuration will not take effect until you enable Proxy ARP.

Use the **show run** command to view the ports on which Local Proxy ARP is enabled.

To enable Local Proxy ARP, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)# ip local-proxy-arp
```

Syntax: [**no**] **ip local-proxy-arp**

Use the **no** form of the command to disable Local Proxy ARP.

Creating static ARP entries

Brocade Layer 3 Switches have a static ARP table, in addition to the regular ARP cache. The static ARP table contains entries that you configure.

Static entries are useful in cases where you want to pre-configure an entry for a device that is not connected to the Layer 3 Switch, or you want to prevent a particular entry from aging out. The software removes a dynamic entry from the ARP cache if the ARP aging interval expires before the entry is refreshed. Static entries do not age out, regardless of whether the Brocade device receives an ARP request from the device that has the entry address.

NOTE
You cannot create static ARP entries on a Layer 2 Switch.

The maximum number of static ARP entries you can configure depends on the software version running on the device. Refer to "Changing the maximum number of entries the static ARP table can hold" on page 40.

To display the ARP cache and static ARP table, refer to the following:

- To display the ARP table, refer to "Displaying the ARP cache" on page 118.
- To display the static ARP table, refer to "Displaying the static ARP table" on page 120.

To create a static ARP entry, enter a command such as the following.

```
Brocade(config)# arp 1 192.168.4.2 0000.0094.2348 ethernet 1/1/2
```

Syntax: **arp** *num ip-addr mac-addr* **ethernet** *port*

The *num* parameter specifies the entry number. You can specify a number from 1 up to the maximum number of static entries allowed on the device.

The *ip-addr* parameter specifies the IP address of the device that has the MAC address of the entry.

The *mac-addr* parameter specifies the MAC address of the entry.

The **ethernet** *port* command specifies the port number attached to the device that has the MAC address of the entry.Specify the *port* variable in the format *stack-unit/slotnum/portnum*.

Changing the maximum number of entries the static ARP table can hold

If you need to change the maximum number of entries supported on a Layer 3 Switch, use the method described in this section.

NOTE
The basic procedure for changing the static ARP table size is the same as the procedure for changing other configurable cache or table sizes. Refer to the section "Displaying system parameter default values" in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

To increase the maximum number of static ARP table entries you can configure on a Brocade Layer 3 Switch, enter commands such as the following at the global CONFIG level of the CLI.

```
Brocade(config)# system-max ip-static-arp 1000
Brocade(config)# write memory
Brocade(config)# end
Brocade# reload
```

NOTE
You must save the configuration to the startup-config file and reload the software after changing the static ARP table size to place the change into effect.

Syntax: **system-max ip-static-arp** *num*

The *num* parameter indicates the maximum number of static ARP entriesdepending on the software version running on the device.

Configuring forwarding parameters

The following configurable parameters control the forwarding behavior of Brocade Layer 3 Switches:

- Time-To-Live (TTL) threshold
- Forwarding of directed broadcasts
- Forwarding of source-routed packets
- Ones-based and zero-based broadcasts

All these parameters are global and thus affect all IP interfaces configured on the Layer 3 Switch.

To configure these parameters, use the procedures in the following sections.

Changing the TTL threshold

The time to live (TTL) threshold prevents routing loops by specifying the maximum number of router hops an IP packet originated by the Layer 3 Switch can travel through. Each device capable of forwarding IP that receives the packet decrements (decreases) the packet TTL by one. If a device receives a packet with a TTL of 1 and reduces the TTL to zero, the device drops the packet.

The default TTL is 64. You can change the TTL to a value from 1 through 255.

To modify the TTL threshold to 25, enter the **ip ttl** command.

```
Brocade(config)# ip ttl 25
```

Syntax: **ip ttl** *1-255*

Enabling forwarding of directed broadcasts

A directed broadcast is an IP broadcast to all devices within a single directly-attached network or subnet. A net-directed broadcast goes to all devices on a given network. A subnet-directed broadcast goes to all devices within a given subnet.

NOTE
A less common type, the all-subnets broadcast, goes to all directly-attached subnets. Forwarding for this broadcast type also is supported, but most networks use IP multicasting instead of all-subnet broadcasting.

Forwarding for all types of IP directed broadcasts is disabled by default. You can enable forwarding for all types if needed. You cannot enable forwarding for specific broadcast types.

To enable forwarding of IP directed broadcasts, enter the **ip directed-broadcast** command.

```
Brocade(config)# ip directed-broadcast
```

Syntax: [**no**] **ip directed-broadcast**

Brocade software makes the forwarding decision based on the router's knowledge of the destination network prefix. Routers cannot determine that a message is unicast or directed broadcast apart from the destination network prefix. The decision to forward or not forward the message is by definition only possible in the last hop router.

To disable the directed broadcasts, enter the **no ip directed-broadcast** command in the CONFIG mode.

```
Brocade(config)# no ip directed-broadcast
```

To enable directed broadcasts on an individual interface instead of globally for all interfaces, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)# ip directed-broadcast
```

Syntax: [**no**] **ip directed-broadcast**

Disabling forwarding of IP source-routed packets

A source-routed packet specifies the exact router path for the packet. The packet specifies the path by listing the IP addresses of the router interfaces through which the packet must pass on its way to the destination. The Layer 3 Switch supports both types of IP source routing:

- Strict source routing – requires the packet to pass through only the listed routers. If the Layer 3 Switch receives a strict source-routed packet but cannot reach the next hop interface specified by the packet, the Layer 3 Switch discards the packet and sends an ICMP Source-Route-Failure message to the sender.

  NOTE
  The Layer 3 Switch allows you to disable sending of the Source-Route-Failure messages. Refer to "Disabling ICMP messages" on page 43.

- Loose source routing – requires that the packet pass through all of the listed routers but also allows the packet to travel through other routers, which are not listed in the packet.

The Layer 3 Switch forwards both types of source-routed packets by default. To disable the feature, use either of the following methods. You cannot enable or disable strict or loose source routing separately.

To disable forwarding of IP source-routed packets, enter the **no ip source-route** command.

```
Brocade(config)# no ip source-route
```

Syntax: [**no**] **ip source-route**

To re-enable forwarding of source-routed packets, enter the **ip source-route** command.

```
Brocade(config)# ip source-route
```

Enabling support for zero-based IP subnet broadcasts

By default, the Layer 3 Switch treats IP packets with all ones in the host portion of the address as IP broadcast packets. For example, the Layer 3 Switch treats IP packets with 192.168.22.255/24 as the destination IP address as IP broadcast packets and forwards the packets to all IP hosts within the 192.168.22.x subnet (except the host that sent the broadcast packet to the Layer 3 Switch).

Most IP hosts are configured to receive IP subnet broadcast packets with all ones in the host portion of the address. However, some older IP hosts instead expect IP subnet broadcast packets that have all zeros instead of all ones in the host portion of the address. To accommodate this type of host, you can enable the Layer 3 Switch to treat IP packets with all zeros in the host portion of the destination IP address as broadcast packets.

NOTE
When you enable the Layer 3 Switch for zero-based subnet broadcasts, the Layer 3 Switch still treats IP packets with all ones the host portion as IP subnet broadcasts too. Thus, the Layer 3 Switch can be configured to support all ones only (the default) or all ones *and* all zeroes.

NOTE
This feature applies only to IP subnet broadcasts, not to local network broadcasts. The local network broadcast address is still expected to be all ones.

To enable the Layer 3 Switch for zero-based IP subnet broadcasts in addition to ones-based IP subnet broadcasts, enter the following command.

```
Brocade(config)# ip broadcast-zero
Brocade(config)# write memory
Brocade(config)# end
Brocade# reload
```

NOTE

You must save the configuration and reload the software to place this configuration change into effect.

Syntax: [no] ip broadcast-zero

Disabling ICMP messages

Brocade devices are enabled to reply to ICMP echo messages and send ICMP Destination Unreachable messages by default.

You can selectively disable the following types of Internet Control Message Protocol (ICMP) messages:

- Echo messages (ping messages) – The Layer 3 Switch replies to IP pings from other IP devices.
- Destination Unreachable messages – If the Layer 3 Switch receives an IP packet that it cannot deliver to its destination, the Layer 3 Switch discards the packet and sends a message back to the device that sent the packet to the Layer 3 Switch. The message informs the device that the destination cannot be reached by the Layer 3 Switch.

Disabling replies to broadcast ping requests

By default, Brocade devices are enabled to respond to broadcast ICMP echo packets, which are ping requests.

To disable response to broadcast ICMP echo packets (ping requests), enter the following command.

```
Brocade(config)# no ip icmp echo broadcast-request
```

Syntax: [no] ip icmp echo broadcast-request

If you need to re-enable response to ping requests, enter the following command.

```
Brocade(config)# ip icmp echo broadcast-request
```

Disabling ICMP destination unreachable messages

By default, when a Brocade device receives an IP packet that the device cannot deliver, the device sends an ICMP Unreachable message back to the host that sent the packet. You can selectively disable a Brocade device response to the following types of ICMP Unreachable messages:

- Administration – The packet was dropped by the Brocade device due to a filter or ACL configured on the device.
- Fragmentation-needed – The packet has the Do not Fragment bit set in the IP Flag field, but the Brocade device cannot forward the packet without fragmenting it.
- Host – The destination network or subnet of the packet is directly connected to the Brocade device, but the host specified in the destination IP address of the packet is not on the network.
- Port – The destination host does not have the destination TCP or UDP port specified in the packet. In this case, the host sends the ICMP Port Unreachable message to the Brocade device, which in turn sends the message to the host that sent the packet.
- Protocol – The TCP or UDP protocol on the destination host is not running. This message is different from the Port Unreachable message, which indicates that the protocol is running on the host but the requested protocol port is unavailable.

- **Source-route-failure** – The device received a source-routed packet but cannot locate the next-hop IP address indicated in the packet Source-Route option.

You can disable the Brocade device from sending these types of ICMP messages on an individual basis. To do so, use the following CLI method.

NOTE
Disabling an ICMP Unreachable message type does not change the Brocade device ability to forward packets. Disabling ICMP Unreachable messages prevents the device from generating or forwarding the Unreachable messages.

To disable all ICMP Unreachable messages, enter the **no ip icmp unreachable** command.

```
Brocade(config)# no ip icmp unreachable
```

Syntax: [**no**] **ip icmp unreachable** [**host** | **protocol** | **administration** | **fragmentation-needed** | **port** | **source-route-fail**]

- If you enter the command without specifying a message type (as in the example above), all types of ICMP Unreachable messages listed above are disabled. If you want to disable only specific types of ICMP Unreachable messages, you can specify the message type. To disable more than one type of ICMP message, enter the **no ip icmp unreachable** command for each messages type.
- The **administration** parameter disables ICMP Unreachable (caused by Administration action) messages.
- The **fragmentation-needed** parameter disables ICMP Fragmentation-Needed But Do not-Fragment Bit Set messages.
- The **host** parameter disables ICMP Host Unreachable messages.
- The **port** parameter disables ICMP Port Unreachable messages.
- The **protocol** parameter disables ICMP Protocol Unreachable messages.
- The **source-route-fail** parameter disables ICMP Unreachable (caused by Source-Route-Failure) messages.

To disable ICMP Host Unreachable messages but leave the other types of ICMP Unreachable messages enabled, enter the following commands instead of the command shown above.

```
Brocade(config)# no ip icmp unreachable host
```

If you have disabled all ICMP Unreachable message types but you want to re-enable certain types, for example ICMP Host Unreachable messages, you can do so by entering the following command.

```
Brocade(config)# ip icmp unreachable host
```

Disabling ICMP redirect messages

You can disable or re-enable ICMP redirect messages. By default, a Brocade Layer 3 Switch sends an ICMP redirect message to the source of a misdirected packet in addition to forwarding the packet to the appropriate router. You can disable ICMP redirect messages on a global basis or on an individual port basis.

NOTE
The device forwards misdirected traffic to the appropriate router, even if you disable the redirect messages.

To disable ICMP redirect messages globally, enter the following command at the global CONFIG level of the CLI:

```
Brocade(config)# no ip icmp redirect
```

Syntax: [**no**] **ip icmp redirects**

To disable ICMP redirect messages on a specific interface, enter the following command at the configuration level for the interface:

```
Brocade(config)# interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)# no ip redirect
```

Syntax: [**no**] **ip redirect**

Static routes configuration

The IP route table can receive routes from the following sources:

- **Directly-connected networks** – When you add an IP interface, the Layer 3 Switch automatically creates a route for the network the interface is in.
- **RIP** – If RIP is enabled, the Layer 3 Switch can learn about routes from the advertisements other RIP routers send to the Layer 3 Switch. If the route has a lower administrative distance than any other routes from different sources to the same destination, the Layer 3 Switch places the route in the IP route table.
- **OSPF** – Refer to RIP, but substitute "OSPF" for "RIP".
- **BGP4** – Refer to RIP, but substitute "BGP4" for "RIP".
- **Default network route** – A statically configured default route that the Layer 3 Switch uses if other default routes to the destination are not available. Refer to "Configuring a default network route" on page 54.
- **Statically configured route** – You can add routes directly to the route table. When you add a route to the IP route table, you are creating a static IP route. This section describes how to add static routes to the IP route table.

Static route types

You can configure the following types of static IP routes:

- **Standard** – the static route consists of the destination network address and network mask, and the IP address of the next-hop gateway. You can configure multiple standard static routes with the same metric for load sharing or with different metrics to provide a primary route and backup routes.
- **Interface-based** – the static route consists of the destination network address and network mask, and the Layer 3 Switch interface through which you want the Layer 3 Switch to send traffic for the route. Typically, this type of static route is for directly attached destination networks.
- **Null** – the static route consists of the destination network address and network mask, and the "null0" parameter. Typically, the null route is configured as a backup route for discarding traffic if the primary route is unavailable.

Static IP route parameters

When you configure a static IP route, you must specify the following parameters:

- The IP address and network mask for the route destination network.
- The route path, which can be one of the following:
    - The IP address of a next-hop gateway
    - An Ethernet port
    - A virtual interface (a routing interface used by VLANs for routing Layer 3 protocol traffic among one another)
    - A “null” interface. The Layer 3 Switch drops traffic forwarded to the null interface.

You also can specify the following optional parameters:

- The metric for the route – The value the Layer 3 Switch uses when comparing this route to other routes in the IP route table to the same destination. The metric applies only to routes that the Layer 3 Switch has already placed in the IP route table. The default metric for static IP routes is 1.
- The administrative distance for the route – The value that the Layer 3 Switch uses to compare this route with routes from other route sources to the same destination before placing a route in the IP route table. This parameter does not apply to routes that are already in the IP route table. The default administrative distance for static IP routes is 1.

The default metric and administrative distance values ensure that the Layer 3 Switch always prefers static IP routes over routes from other sources to the same destination.

Multiple static routes to the same destination provide load sharing and redundancy

You can add multiple static routes for the same destination network to provide one or more of the following benefits:

- **IP load balancing** – When you add multiple IP static routes for the same destination to different next-hop gateways, and the routes each have the same metric and administrative distance, the Layer 3 Switch can load balance traffic to the routes’ destination. For information about IP load balancing, refer to “Configuring IP load sharing” on page 55.
- **Path redundancy** – When you add multiple static IP routes for the same destination, but give the routes different metrics or administrative distances, the Layer 3 Switch uses the route with the lowest administrative distance by default, but uses another route to the same destination if the first route becomes unavailable.

Refer to the following sections for examples and configuration information:

- “Configuring load balancing and redundancy using multiple static routes to the same destination” on page 49
- “Configuring standard static IP routes and interface or null static routes to the same destination” on page 50

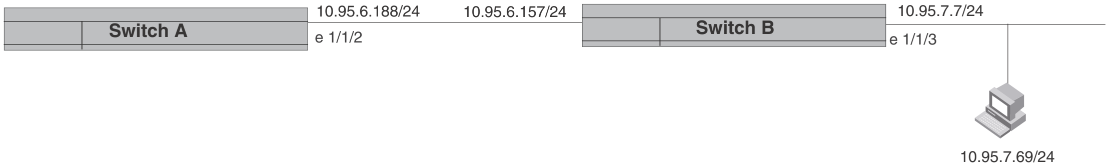
Static route states follow port states

IP static routes remain in the IP route table only so long as the port or virtual interface used by the route is available. If the port or virtual routing interface becomes unavailable, the software removes the static route from the IP route table. If the port or virtual routing interface becomes available again later, the software adds the route back to the route table.

This feature allows the Layer 3 Switch to adjust to changes in network topology. The Layer 3 Switch does not continue trying to use routes on unavailable paths but instead uses routes only when their paths are available.

Figure 4 shows an example of a network containing a static route. The static route is configured on Switch A, as shown in the CLI example following the figure.

FIGURE 4 **Example of a static route**



The following command configures a static route to 10.95.7.0, using 10.95.6.157 as the next-hop gateway.

```
Brocade(config)# ip route 10.95.7.0/24 10.95.6.157
```

When you configure a static IP route, you specify the destination address for the route and the next-hop gateway or Layer 3 Switch interface through which the Layer 3 Switch can reach the route. The Layer 3 Switch adds the route to the IP route table. In this case, Switch A knows that 10.95.6.157 is reachable through port 1/1/2, and also assumes that local interfaces within that subnet are on the same port. Switch A deduces that IP interface 10.95.7.188 is also on port 1/1/2.

The software automatically removes a static IP route from the IP route table if the port used by that route becomes unavailable. When the port becomes available again, the software automatically re-adds the route to the IP route table.

Configuring a static IP route

To configure an IP static route with a destination address of 192.168.0.0 255.0.0.0 and a next-hop router IP address of 192.168.1.1, enter a command such as the following.

```
Brocade(config)# ip route 192.168.0.0 255.0.0.0 192.168.1.1
```

To configure a static IP route with an Ethernet port instead of a next-hop address, enter a command such as the following.

```
Brocade(config)# ip route 192.168.2.69 255.255.255.0 ethernet 1/1/4
```

The command in the previous example configures a static IP route for destination network 192.168.2.69/24. Since an Ethernet port is specified instead of a gateway IP address as the next hop, the Layer 3 Switch always forwards traffic for the 192.168.2.69/24 network to port 1/1/4. The command in the following example configures an IP static route that uses virtual interface 3 as its next hop.

```
Brocade(config)# ip route 192.168.2.71 255.255.255.0 ve 3
```

The command in the following example configures an IP static route that uses port 1/1/2 as its next hop.

```
Brocade(config)# ip route 192.168.2.73 255.255.255.0 ethernet 1/1/2
```

Syntax: **ip route** *dest-ip-addr dest-mask*
next-hop-ip-addr |
ethernet *stack-unit/slotnum/portnum* | **ve** *num*
[*metric*] [**distance** *num*]

or

Syntax: **ip route** *dest-ip-addr/mask-bits*
next-hop-ip-addr |
ethernet *stack-unit/slotnum/portnum* | **ve** *num*
[*metric*] [**distance** *num*]

The *dest-ip-addr* is the route destination. The *dest-mask* is the network mask for the route destination IP address. Alternatively, you can specify the network mask information by entering a forward slash followed by the number of bits in the network mask. For example, you can enter 192.168.0.0 255.255.255.0 as 192.168.0.0/.24.

The *next-hop-ip-addr* is the IP address of the next-hop router (gateway) for the route.

If you do not want to specify a next-hop IP address, you can instead specify a port or interface number on the Layer 3 Switch. The *num* parameter is a virtual interface number. If you instead specify an Ethernet port, the *portnum* is the port number (including the stack unit and slot number). In this case, the Layer 3 Switch forwards packets destined for the static route destination network to the specified interface. Conceptually, this feature makes the destination network like a directly connected network, associated with a specific Layer 3 Switch interface.

NOTE
The port or virtual interface you use for the static route next hop must have at least one IP address configured on it. The address does not need to be in the same subnet as the destination network.

The *metric* parameter can be a number from 1 through 16. The default is 1.

NOTE
If you specify 16, RIP considers the metric to be infinite and thus also considers the route to be unreachable.

The **distance** *num* parameter specifies the administrative distance of the route. When comparing otherwise equal routes to a destination, the Layer 3 Switch prefers lower administrative distances over higher ones, so make sure you use a low value for your default route. The default is 1.

NOTE
The Layer 3 Switch will replace the static route if the it receives a route with a lower administrative distance. Refer to "Administrative distance" on page 207 for a list of the default administrative distances for all types of routes.

NOTE
You can also assign the default router as the destination by entering 0.0.0.0 0.0.0.0 xxx.xxx.xxx.xxx.

Configuring a "Null" route

You can configure the Layer 3 Switch to drop IP packets to a specific network or host address by configuring a "null" (sometimes called "null0") static route for the address. When the Layer 3 Switch receives a packet destined for the address, the Layer 3 Switch drops the packet instead of forwarding it.

To configure a null static route, use the following CLI method.

To configure a null static route to drop packets destined for network 192.168.22.x, enter the following commands.

```
Brocade(config)# ip route 192.168.22.0 255.255.255.0 null0
Brocade(config)# write memory
```

Syntax: **ip route** *ip-addr ip-mask* **null0** [*metric*] [**distance** *num*]

or

Syntax: **ip route** *ip-addr/mask-bits* **null0** [*metric*] [**distance** *num*]

To display the maximum value for your device, enter the **show default values** command. The maximum number of static IP routes the system can hold is listed in the ip-static-route row in the System Parameters section of the display. To change the maximum value, use the **system-max ip-static-route** *num* command at the global CONFIG level.

The *ip-addr* parameter specifies the network or host address. The Layer 3 Switch will drop packets that contain this address in the destination field instead of forwarding them.

The *ip-mask* parameter specifies the network mask. Ones are significant bits and zeros allow any value. For example, the mask 255.255.255.0 matches on all hosts within the Class C subnet address specified by *ip-addr*. Alternatively, you can specify the number of bits in the network mask. For example, you can enter 192.168.22.0/24 instead of 192.168.22.0 255.255.255.0.

The **null0** parameter indicates that this is a null route. You must specify this parameter to make this a null route.

The *metric* parameter adds a cost to the route. You can specify from 1 through 16. The default is 1.

The distance *num* parameter configures the administrative distance for the route. You can specify a value from 1 through 255. The default is 1. The value 255 makes the route unusable.

NOTE
The last two parameters are optional and do not affect the null route, unless you configure the administrative distance to be 255. In this case, the route is not used and the traffic might be forwarded instead of dropped.

Configuring load balancing and redundancy using multiple static routes to the same destination

You can configure multiple static IP routes to the same destination, for the following benefits:

- **IP load sharing** – If you configure more than one static route to the same destination, and the routes have different next-hop gateways but have the same metrics, the Layer 3 Switch load balances among the routes using basic round-robin. For example, if you configure two static routes with the same metrics but to different gateways, the Layer 3 Switch alternates between the two routes. For information about IP load balancing, refer to "Configuring IP load sharing" on page 55.

- **Backup Routes** – If you configure multiple static IP routes to the same destination, but give the routes different next-hop gateways and different metrics, the Layer 3 Switch will always use the route with the lowest metric. If this route becomes unavailable, the Layer 3 Switch will fail over to the static route with the next-lowest metric, and so on.

NOTE
You also can bias the Layer 3 Switch to select one of the routes by configuring them with different administrative distances. However, make sure you do not give a static route a higher administrative distance than other types of routes, unless you want those other types to be preferred over the static route. For a list of the default administrative distances, refer to "Administrative distance" on page 207.

The steps for configuring the static routes are the same as described in the previous section. The following sections provide examples.

To configure multiple static IP routes, enter commands such as the following.

```
Brocade(config)# ip route 192.168.2.69 255.255.255.0 192.168.22.1
Brocade(config)# ip route 192.168.2.69 255.255.255.0 192.168.10.1
```

The commands in the previous example configure two static IP routes. The routes go to different next-hop gateways but have the same metrics. These commands use the default metric value (1), so the metric is not specified. These static routes are used for load sharing among the next-hop gateways.

The following commands configure static IP routes to the same destination, but with different metrics. The route with the lowest metric is used by default. The other routes are backups in case the first route becomes unavailable. The Layer 3 Switch uses the route with the lowest metric if the route is available.

```
Brocade(config)# ip route 192.168.2.69 255.255.255.0 192.168.22.1
Brocade(config)# ip route 192.168.2.69 255.255.255.0 192.168.10.1 2
Brocade(config)# ip route 192.168.2.69 255.255.255.0 192.168.1 3
```

In this example, each static route has a different metric. The metric is not specified for the first route, so the default (1) is used. A metric is specified for the second and third static IP routes. The second route has a metric of two and the third route has a metric of 3. Thus, the second route is used only of the first route (which has a metric of 1) becomes unavailable. Likewise, the third route is used only if the first and second routes (which have lower metrics) are both unavailable.

For complete syntax information, refer to "Configuring a static IP route" on page 47.

Configuring standard static IP routes and interface or null static routes to the same destination

You can configure a null0 or interface-based static route to a destination and also configure a normal static route to the same destination, so long as the route metrics are different.

When the Layer 3 Switch has multiple routes to the same destination, the Layer 3 Switch always prefers the route with the lowest metric. Generally, when you configure a static route to a destination network, you assign the route a low metric so that the Layer 3 Switch prefers the static route over other routes to the destination.

This feature is especially useful for the following configurations. These are not the only allowed configurations but they are typical uses of this enhancement:

- When you want to ensure that if a given destination network is unavailable, the Layer 3 Switch drops (forwards to the null interface) traffic for that network instead of using alternate paths to route the traffic. In this case, assign the normal static route to the destination network a lower metric than the null route.
- When you want to use a specific interface by default to route traffic to a given destination network, but want to allow the Layer 3 Switch to use other interfaces to reach the destination network if the path that uses the default interface becomes unavailable. In this case, give the interface route a lower metric than the normal static route.

NOTE
You cannot add a null or interface-based static route to a network if there is already a static route of any type with the same metric you specify for the null or interface-based route.

Figure 5 shows an example of two static routes configured for the same destination network. In this example, one of the routes is a standard static route and has a metric of 1. The other static route is a null route and has a higher metric than the standard static route. The Layer 3 Switch always prefers the static route with the lower metric. In this example, the Layer 3 Switch always uses the standard static route for traffic to destination network 192.168.7.0/24, unless that route becomes unavailable, in which case the Layer 3 Switch sends traffic to the null route instead.

FIGURE 5 Standard and null static routes to the same destination network

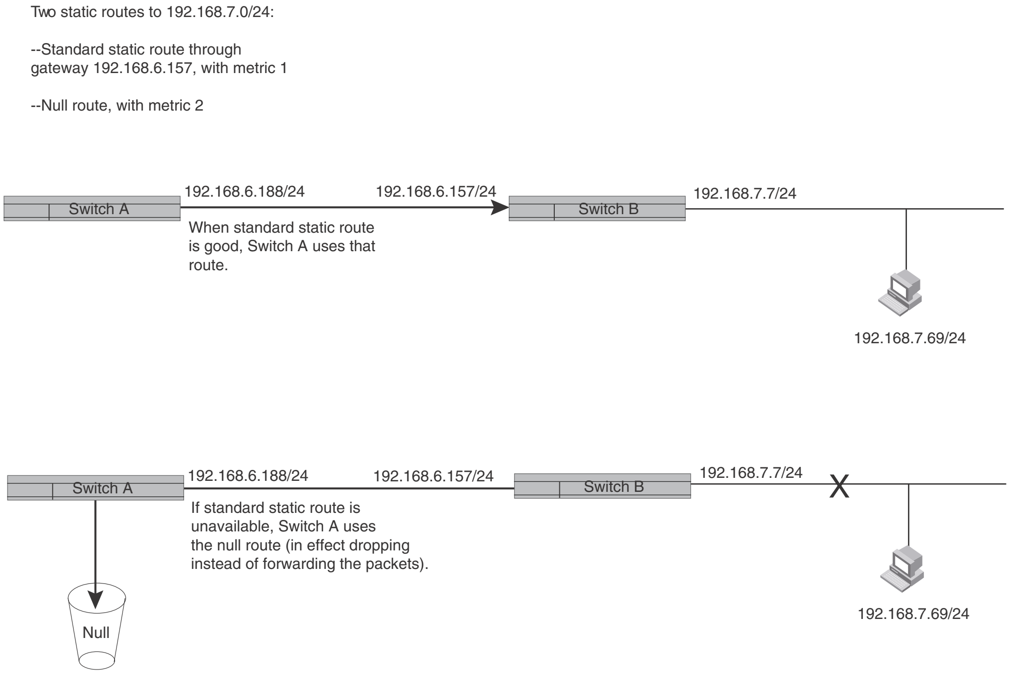
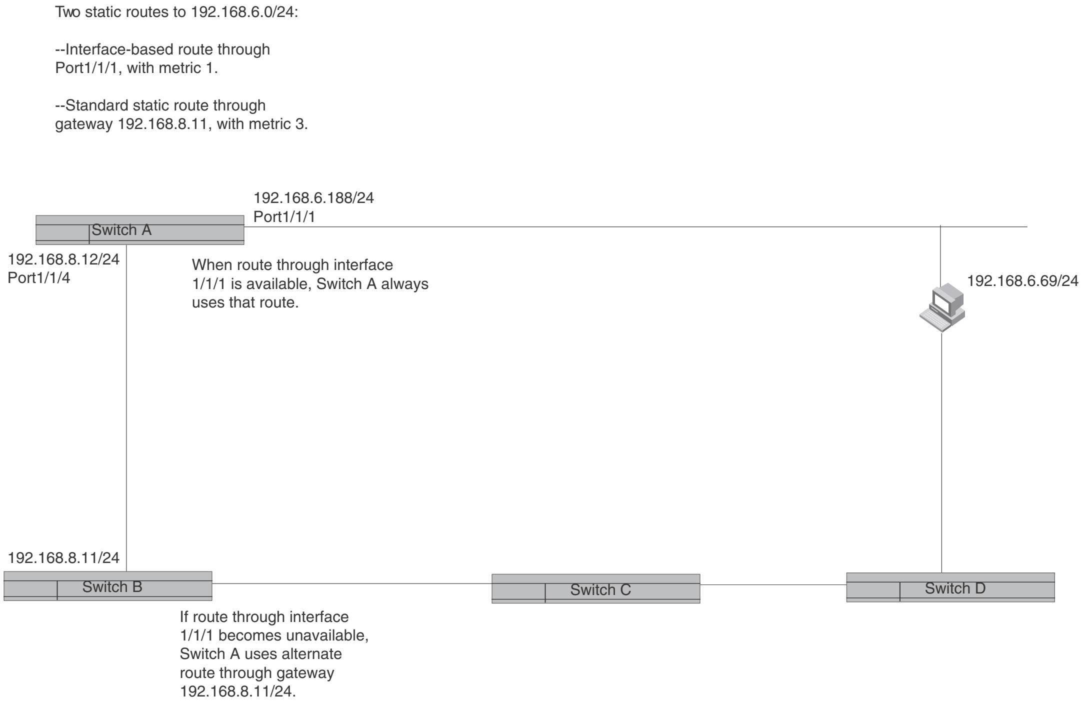

Two static routes to 192.168.7.0/24:
--Standard static route through gateway 192.168.6.157, with metric 1
--Null route, with metric 2
Switch A
192.168.6.188/24
192.168.6.157/24
Switch B
192.168.7.7/24
When standard static route is good, Switch A uses that route.
192.168.7.69/24
Switch A
192.168.6.188/24
192.168.6.157/24
Switch B
192.168.7.7/24
X
If standard static route is unavailable, Switch A uses the null route (in effect dropping instead of forwarding the packets).
Null
192.168.7.69/24

Figure 6 shows another example of two static routes. In this example, a standard static route and an interface-based static route are configured for destination network 192.168.6.0/24. The interface-based static route has a lower metric than the standard static route. As a result, the Layer 3 Switch always prefers the interface-based route when the route is available. However, if the interface-based route becomes unavailable, the Layer 3 Switch still forwards the traffic toward the destination using an alternate route through gateway 192.168.8.11/24.

FIGURE 6 Standard and interface routes to the same destination network


Two static routes to 192.168.6.0/24:
--Interface-based route through Port1/1/1, with metric 1.
--Standard static route through gateway 192.168.8.11, with metric 3.
192.168.6.188/24
Port1/1/1
Switch A
192.168.8.12/24
Port1/1/4
When route through interface 1/1/1 is available, Switch A always uses that route.
192.168.6.69/24
192.168.8.11/24
Switch B
Switch C
Switch D
If route through interface 1/1/1 becomes unavailable, Switch A uses alternate route through gateway 192.168.8.11/24.

To configure a standard static IP route and a null route to the same network as shown in Figure 5 on page 52, enter commands such as the following.

```
Brocade(config)# ip route 192.168.7.0/24 192.168.6.157/24 1
Brocade(config)# ip route 192.168.7.0/24 null0 3
```

The first command configures a standard static route, which includes specification of the next-hop gateway. The command also gives the standard static route a metric of 1, which causes the Layer 3 Switch to always prefer this route when the route is available.

The second command configures another static route for the same destination network, but the second route is a null route. The metric for the null route is 3, which is higher than the metric for the standard static route. If the standard static route is unavailable, the software uses the null route.

For complete syntax information, refer to "Configuring a static IP route" on page 47.

To configure a standard static route and an interface-based route to the same destination, enter commands such as the following.

```
Brocade(config)# ip route 192.168.6.0/24 ethernet 1/1 1
Brocade(config)# ip route 192.168.6.0/24 192.168.8.11/24 3
```

The first command configured an interface-based static route through Ethernet port 1/1/1. The command assigns a metric of 1 to this route, causing the Layer 3 Switch to always prefer this route when it is available. If the route becomes unavailable, the Layer 3 Switch uses an alternate route through the next-hop gateway 192.168.8.11/24.

Configuring a default network route

The Layer 3 Switch enables you to specify a candidate default route without the need to specify the next hop gateway. If the IP route table does not contain an explicit default route (for example, 0.0.0.0/0) or propagate an explicit default route through routing protocols, the software can use the default network route as a default route instead.

When the software uses the default network route, it also uses the default network route's next hop gateway as the gateway of last resort.

This feature is especially useful in environments where network topology changes can make the next hop gateway unreachable. This feature allows the Layer 3 Switch to perform default routing even if the default network route's default gateway changes.

The feature thus differs from standard default routes. When you configure a standard default route, you also specify the next hop gateway. If a topology change makes the gateway unreachable, the default route becomes unusable.

For example, if you configure 10.10.10.0/24 as a candidate default network route, if the IP route table does not contain an explicit default route (0.0.0.0/0), the software uses the default network route and automatically uses that route's next hop gateway as the default gateway. If a topology change occurs and as a result the default network route's next hop gateway changes, the software can still use the default network route. To configure a default network route, use the following CLI method.

If you configure more than one default network route, the Layer 3 Switch uses the following algorithm to select one of the routes.

1. Use the route with the lowest administrative distance.
2. If the administrative distances are equal:
    - Are the routes from different routing protocols (RIP, OSPF, or BGP4)? If so, use the route with the lowest IP address.
    - If the routes are from the same routing protocol, use the route with the best metric. The meaning of “best” metric depends on the routing protocol:
    - RIP – The metric is the number of hops (additional routers) to the destination. The best route is the route with the fewest hops.
    - OSPF – The metric is the path cost associated with the route. The path cost does not indicate the number of hops but is instead a numeric value associated with each route. The best route is the route with the lowest path cost.
    - BGP4 – The metric is the Multi-exit Discriminator (MED) associated with the route. The MED applies to routes that have multiple paths through the same AS. The best route is the route with the lowest MED.

You can configure up to four default network routes.

To configure a default network route, enter commands such as the following.

```
Brocade(config)# ip default-network 192.168.22.0
Brocade(config)# write memory
```

Syntax: **ip default-network** *ip-addr*

The *ip-addr* parameter specifies the network address.

To verify that the route is in the route table, enter the following command at any level of the CLI.

```
Brocade# show ip route
Total number of IP routes: 2
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
      Destination       NetMask          Gateway          Port   Cost   Type
1     10.157.20.0      255.255.255.0     0.0.0.0           lb1    1      D
2     10.157.22.0      255.255.255.0     0.0.0.0           1/1/1   1      *D
```

This example shows two routes. Both of the routes are directly attached, as indicated in the Type column. However, one of the routes is shown as type "*D", with an asterisk (*). The asterisk indicates that this route is a candidate default network route.

Configuring IP load sharing

The IP route table can contain more than one path to a given destination. When this occurs, the Layer 3 Switch selects the path with the lowest cost as the path for forwarding traffic to the destination. If the IP route table contains more than one path to a destination and the paths each have the lowest cost, then the Layer 3 Switch uses *IP load sharing* to select a path to the destination.[1]

IP load sharing uses a hashing algorithm based on the source IP address, destination IP address, and protocol field in the IP header, TCP, and UDP information.

NOTE
IP load sharing is based on next-hop routing, and not on source routing.

NOTE
The term "path" refers to the next-hop router to a destination, not to the entire route to a destination. Thus, when the software compares multiple equal-cost paths, the software is comparing paths that use different next-hop routers, with equal costs, to the same destination.

In many contexts, the terms "route" and "path" mean the same thing. Most of the user documentation uses the term "route" throughout. The term "path" is used in this section to refer to an individual next-hop router to a destination, while the term "route" refers collectively to the multiple paths to the destination. Load sharing applies when the IP route table contains multiple, equal-cost paths to a destination.

NOTE
Brocade devices also perform load sharing among the ports in aggregate links. Refer to the section "Trunk group load sharing" in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

How multiple equal-cost paths enter the IP route table

IP load sharing applies to equal-cost paths in the IP route table. Routes that are eligible for load sharing can enter the table from any of the following sources:

- IP static routes
- Routes learned through RIP
- Routes learned through OSPF

[1] IP load sharing is also called "Equal-Cost Multi-Path (ECMP)" load sharing or just "ECMP"

- Routes learned through BGP4

Administrative distance for each IP route

The administrative distance is a unique value associated with each type (source) of IP route. Each path has an administrative distance. The administrative distance is not used when performing IP load sharing, but the administrative distance is used when evaluating multiple equal-cost paths to the same destination from different sources, such as RIP, OSPF and so on.

The value of the administrative distance is determined by the source of the route. The Layer 3 Switch is configured with a unique administrative distance value for each IP route source.

When the software receives multiple paths to the same destination and the paths are from different sources, the software compares the administrative distances of the paths and selects the path with the lowest distance. The software then places the path with the lowest administrative distance in the IP route table. For example, if the Layer 3 Switch has a path learned from OSPF and a path learned from RIP for a given destination, only the path with the lower administrative distance enters the IP route table.

Here are the default administrative distances on the Brocade Layer 3 Switch:

- Directly connected – 0 (this value is not configurable)
- Static IP route – 1 (applies to all static routes, including default routes and default network routes)
- External Border Gateway Protocol eBGP) – 20
- OSPF – 110
- RIP – 120
- Internal Gateway Protocol (iBGP) – 200
- Unknown – 255 (the router will not use this route)

Lower administrative distances are preferred over higher distances. For example, if the router receives routes for the same network from OSPF and from RIP, the router will prefer the OSPF route by default.

NOTE
You can change the administrative distances individually. Refer to the configuration chapter for the route source for information.

Since the software selects only the path with the lowest administrative distance, and the administrative distance is determined by the path source, IP load sharing does not apply to paths from different route sources. IP load sharing applies only when the IP route table contains multiple paths to the same destination, from the same IP route source.

IP load sharing does not apply to paths that come from different sources.

Path cost

The cost parameter provides a common basis of comparison for selecting from among multiple paths to a given destination. Each path in the IP route table has a cost. When the IP route table contains multiple paths to a destination, the Layer 3 Switch chooses the path with the lowest cost. When the IP route table contains more than one path with the lowest cost to a destination, the Layer 3 Switch uses IP load sharing to select one of the lowest-cost paths.

The source of a path cost value depends on the source of the path:

- **IP static route** – The value you assign to the metric parameter when you configure the route. The default metric is 1. Refer to "Configuring load balancing and redundancy using multiple static routes to the same destination" on page 49.
- **RIP** – The number of next-hop routers to the destination.
- **OSPF** – The Path Cost associated with the path. The paths can come from any combination of inter-area, intra-area, and external Link State Advertisements (LSAs).
- **BGP4** – The path Multi-Exit Discriminator (MED) value.

NOTE
If the path is redistributed between two or more of the above sources before entering the IP route table, the cost can increase during the redistribution due to settings in redistribution filters.

Static route, OSPF, and BGP4 load sharing

IP load sharing and load sharing for static routes, OSPF routes, and BGP4 routes are individually configured. Multiple equal-cost paths for a destination can enter the IP route table only if the source of the paths is configured to support multiple equal-cost paths. For example, if BGP4 allows only one path with a given cost for a given destination, the BGP4 route table cannot contain equal-cost paths to the destination. Consequently, the IP route table will not receive multiple equal-cost paths from BGP4.

Table 6 lists the default and configurable maximum numbers of paths for each IP route source that can provide equal-cost paths to the IP route table. The table also lists where to find configuration information for the route source load sharing parameters.

The load sharing state for all the route sources is based on the state of IP load sharing. Since IP load sharing is enabled by default on all Brocade Layer 3 Switches, load sharing for static IP routes, RIP routes, OSPF routes, and BGP4 routes also is enabled by default.

TABLE 6 **Default load sharing parameters for route sources**

Route source	Default maximum number of paths	Maximum number of paths	See...
Static IP route	4[1]	8[1]	page 58
RIP	4[1]	8[1]	page 58
OSPF	4	8	page 58
BGP4	1	4	page 291

[1] This value depends on the value for IP load sharing, and is not separately configurable.

How IP load sharing works

When the Layer 3 Switch receives traffic for a destination and the IP route table contains multiple, equal-cost paths to that destination, the device checks the IP forwarding cache for a forwarding entry for the destination. The IP forwarding cache provides a fast path for forwarding IP traffic, including load-balanced traffic. The cache contains entries that associate a destination host or network with a path (next-hop router).

- If the IP forwarding sharing cache contains a forwarding entry for the destination, the device uses the entry to forward the traffic.

- If the IP load forwarding cache does not contain a forwarding entry for the destination, the software selects a path from among the available equal-cost paths to the destination, then creates a forwarding entry in the cache based on the calculation. Subsequent traffic for the same destination uses the forwarding entry.

Response to path state changes

If one of the load-balanced paths to a cached destination becomes unavailable, or the IP route table receives a new equal-cost path to a cached destination, the software removes the unavailable path from the IP route table. Then the software selects a new path.

Disabling or re-enabling load sharing

To disable IP load sharing, enter the following commands.

```
Brocade(config)# no ip load-sharing
```

Syntax: [**no**] **ip load-sharing**

Changing the maximum number of ECMP (load sharing) paths

You can change the maximum number of paths the Layer 3 Switch supports to a value from 2 through 8. The maximum number of ECMP load sharing paths supported per device is 8.

For optimal results, set the maximum number of paths to a value at least as high as the maximum number of equal-cost paths your network typically contains. For example, if the Layer 3 Switch you are configuring for IP load sharing has six next-hop routers, set the maximum paths value to six.

NOTE
If the setting for the maximum number of paths is lower than the actual number of equal-cost paths, the software does not use all the paths for load sharing for RIP routes. Run the **clear ip route** command to fix this issue.

To change the number of IP load sharing paths, enter a command such as the following.

```
Brocade(config)# ip load-sharing 6
```

Syntax: [**no**] **ip load-sharing** [*num*]

The *num* parameter specifies the number of paths and can be from 2 through 8, depending on the device you are configuring.

ICMP Router Discovery Protocol configuration

The ICMP Router Discovery Protocol (IRDP) is used by Brocade Layer 3 Switches to advertise the IP addresses of its router interfaces to directly attached hosts. IRDP is disabled by default. You can enable the feature on a global basis or on an individual port basis:

- If you enable the feature globally, all ports use the default values for the IRDP parameters.
- If you leave the feature disabled globally but enable it on individual ports, you also can configure the IRDP parameters on an individual port basis.

NOTE
You can configure IRDP parameters only an individual port basis. To do so, IRDP must be disabled globally and enabled only on individual ports. You cannot configure IRDP parameters if the feature is globally enabled.

When IRDP is enabled, the Layer 3 Switch periodically sends Router Advertisement messages out the IP interfaces on which the feature is enabled. The messages advertise the Layer 3 Switch IP addresses to directly attached hosts who listen for the messages. In addition, hosts can be configured to query the Layer 3 Switch for the information by sending Router Solicitation messages.

Some types of hosts use the Router Solicitation messages to discover their default gateway. When IRDP is enabled on the Brocade Layer 3 Switch, the Layer 3 Switch responds to the Router Solicitation messages. Some clients interpret this response to mean that the Layer 3 Switch is the default gateway. If another router is actually the default gateway for these clients, leave IRDP disabled on the Brocade Layer 3 Switch.

IRDP parameters

IRDP uses the following parameters. If you enable IRDP on individual ports instead of enabling the feature globally, you can configure these parameters on an individual port basis:

- Packet type – The Layer 3 Switch can send Router Advertisement messages as IP broadcasts or as IP multicasts addressed to IP multicast group 224.0.0.1. The packet type is IP broadcast.
- Maximum message interval and minimum message interval – When IRDP is enabled, the Layer 3 Switch sends the Router Advertisement messages every 450 – 600 seconds by default. The time within this interval that the Layer 3 Switch selects is random for each message and is not affected by traffic loads or other network factors. The random interval minimizes the probability that a host will receive Router Advertisement messages from other routers at the same time. The interval on each IRDP-enabled Layer 3 Switch interface is independent of the interval on other IRDP-enabled interfaces. The default maximum message interval is 600 seconds. The default minimum message interval is 450 seconds.
- Hold time – Each Router Advertisement message contains a hold time value. This value specifies the maximum amount of time the host should consider an advertisement to be valid until a newer advertisement arrives. When a new advertisement arrives, the hold time is reset. The hold time is always longer than the maximum advertisement interval. Therefore, if the hold time for an advertisement expires, the host can reasonably conclude that the router interface that sent the advertisement is no longer available. The default hold time is three times the maximum message interval.
- Preference – If a host receives multiple Router Advertisement messages from different routers, the host selects the router that sent the message with the highest preference as the default gateway. The preference can be a number from 0-4294967296 to 0-4294967295. The default is 0.

Enabling IRDP globally

To globally enable IRDP, enter the following command.

```
Brocade(config)# ip irdp
```

This command enables IRDP on the IP interfaces on all ports. Each port uses the default values for the IRDP parameters. The parameters are not configurable when IRDP is globally enabled.

Enabling IRDP on an individual port

To enable IRDP on an individual interface and change IRDP parameters, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# ip irdp maxadvertinterval 400
```

This example shows how to enable IRDP on a specific port and change the maximum advertisement interval for Router Advertisement messages to 400 seconds.

NOTE
To enable IRDP on individual ports, you must leave the feature globally disabled.

Syntax: [**no**] **ip irdp** [**broadcast** | **multicast**] [**holdtime** *seconds*] [**maxadvertinterval** *seconds*] [**minadvertinterval** *seconds*] [**preference** *number*]

The **broadcast** | **multicast** parameter specifies the packet type the Layer 3 Switch uses to send Router Advertisement:

- **broadcast** – The Layer 3 Switch sends Router Advertisement as IP broadcasts. This is the default.
- **multicast** – The Layer 3 Switch sends Router Advertisement as multicast packets addressed to IP multicast group 224.0.0.1.

The **holdtime** *seconds* parameter specifies how long a host that receives a Router Advertisement from the Layer 3 Switch should consider the advertisement to be valid. When a host receives a new Router Advertisement message from the Layer 3 Switch, the host resets the hold time for the Layer 3 Switch to the hold time specified in the new advertisement. If the hold time of an advertisement expires, the host discards the advertisement, concluding that the router interface that sent the advertisement is no longer available. The value must be greater than the value of the **maxadvertinterval** parameter and cannot be greater than 9000. The default is three times the value of the **maxadvertinterval** parameter.

The **maxadvertinterval** parameter specifies the maximum amount of time the Layer 3 Switch waits between sending Router Advertisements. You can specify a value from 1 to the current value of the **holdtime** parameter. The default is 600 seconds.

The **minadvertinterval** parameter specifies the minimum amount of time the Layer 3 Switch can wait between sending Router Advertisements. The default is three-fourths (0.75) the value of the **maxadvertinterval** parameter. If you change the **maxadvertinterval** parameter, the software automatically adjusts the **minadvertinterval** parameter to be three-fourths the new value of the **maxadvertinterval** parameter. If you want to override the automatically configured value, you can specify an interval from 1 to the current value of the **maxadvertinterval** parameter.

The **preference** *number* parameter specifies the IRDP preference level of this Layer 3 Switch. If a host receives Router Advertisements from multiple routers, the host selects the router interface that sent the message with the highest interval as the host default gateway. The valid range is 0-4294967296 to 0-4294967295. The default is 0.

Reverse Address Resolution Protocol configuration

The Reverse Address Resolution Protocol (RARP) provides a simple mechanism for directly-attached IP hosts to boot over the network. RARP allows an IP host that does not have a means of storing its IP address across power cycles or software reloads to query a directly-attached router for an IP address.

RARP is enabled by default. However, you must create a RARP entry for each host that will use the Layer 3 Switch for booting. A RARP entry consists of the following information:

- The entry number – the entry sequence number in the RARP table.
- The MAC address of the boot client.
- The IP address you want the Layer 3 Switch to give to the client.

When a client sends a RARP broadcast requesting an IP address, the Layer 3 Switch responds to the request by looking in the RARP table for an entry that contains the client MAC address:

- If the RARP table contains an entry for the client, the Layer 3 Switch sends a unicast response to the client that contains the IP address associated with the client MAC address in the RARP table.
- If the RARP table does not contain an entry for the client, the Layer 3 Switch silently discards the RARP request and does not reply to the client.

How RARP differs from BootP and DHCP

RARP and BootP/DHCP are different methods for providing IP addresses to IP hosts when they boot. These methods differ in the following ways:

- Location of configured host addresses:
  - RARP requires static configuration of the host IP addresses on the Layer 3 Switch. The Layer 3 Switch replies directly to a host request by sending an IP address you have configured in the RARP table.
  - The Layer 3 Switch forwards BootP and DHCP requests to a third-party BootP/DHCP server that contains the IP addresses and other host configuration information.
- Connection of host to boot source (Layer 3 Switch or BootP/DHCP server):
  - RARP requires the IP host to be directly attached to the Layer 3 Switch.
  - An IP host and the BootP/DHCP server can be on different networks and on different routers, so long as the routers are configured to forward (“help”) the host boot request to the boot server.
  - You can centrally configure other host parameters on the BootP/DHCP server, in addition to the IP address, and supply those parameters to the host along with its IP address.

To configure the Layer 3 Switch to forward BootP/DHCP requests when boot clients and the boot servers are on different subnets on different Layer 3 Switch interfaces, refer to “BootP and DHCP relay parameter configuration” on page 65.

Disabling RARP

RARP is enabled by default. To disable RARP, enter the following command at the global CONFIG level.

```
Brocade(config)# no ip rarp
```

Syntax: [**no**] **ip rarp**

To re-enable RARP, enter the following command.

```
Brocade(config)# ip rarp
```

Creating static RARP entries

You must configure the RARP entries for the RARP table. The Layer 3 Switch can send an IP address in reply to a client RARP request only if create a RARP entry for that client.

To assign a static IP RARP entry for static routes on a Brocade router, enter a command such as the following.

```
Brocade(config)# rarp 1 0000.0054.2348 192.168.4.2
```

This command creates a RARP entry for a client with MAC address 0000.0054.2348. When the Layer 3 Switch receives a RARP request from this client, the Layer 3 Switch replies to the request by sending IP address 192.168.4.2 to the client.

Syntax: r**arp** *number mac-addr. ip-addr*

The *number* parameter identifies the RARP entry number. You can specify an unused number from 1 to the maximum number of RARP entries supported on the device. To determine the maximum number of entries supported on the device, refer to the section “Displaying and modifying system parameter default settings” in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

The *mac-addr* parameter specifies the MAC address of the RARP client.

The *ip-addr* parameter specifies the IP address the Layer 3 Switch will give the client in response to the client RARP request.

Changing the maximum number of static RARP entries supported

The number of RARP entries the Layer 3 Switch supports depends on how much memory the Layer 3 Switch has. To determine how many RARP entries your Layer 3 Switch can have, display the system default information using the procedure in the section“Displaying and modifying system parameter default settings” in the *Brocade ICX 6650 Platform and Layer 2 Switching Configuration Guide*.

If your Layer 3 Switch allows you to increase the maximum number of RARP entries, you can use a procedure in the same section to do so.

NOTE
You must save the configuration to the startup-config file and reload the software after changing the RARP cache size to place the change into effect.

Configuring UDP broadcast and IP helper parameters

Some applications rely on client requests sent as limited IP broadcasts addressed to the UDP application port. If a server for the application receives such a broadcast, the server can reply to the client. Routers do not forward subnet directed broadcasts, so the client and server must be on the same network for the broadcast to reach the server. If the client and server are on different networks (on opposite sides of a router), the client request cannot reach the server.

You can configure the Layer 3 Switch to forward clients‘ requests to UDP application servers. To do so:

- Enable forwarding support for the UDP application port, if forwarding support is not already enabled.
- Configure a helper adders on the interface connected to the clients. Specify the helper address to be the IP address of the application server or the subnet directed broadcast address for the IP subnet the server is in. A helper address is associated with a specific interface and applies only to client requests received on that interface. The Layer 3 Switch forwards client requests for any of the application ports the Layer 3 Switch is enabled to forward to the helper address.

Forwarding support for the following application ports is enabled by default:

- bootps (port 67)
- dns (port 53)
- tftp (port 69)
- time (port 37)
- netbios-ns (port 137)
- netbios-dgm (port 138)
- tacacs (port 65)

NOTE
The application names are the names for these applications that the Layer 3 Switch software recognizes, and might not match the names for these applications on some third-party devices. The numbers listed in parentheses are the UDP port numbers for the applications. The numbers come from RFC 1340.

NOTE
Forwarding support for BootP/DHCP is enabled by default. If you are configuring the Layer 3 Switch to forward BootP/DHCP requests, refer to “BootP and DHCP relay parameter configuration” on page 65.

You can enable forwarding for other applications by specifying the application port number.

You also can disable forwarding for an application.

NOTE
If you disable forwarding for a UDP application, forwarding of client requests received as broadcasts to helper addresses is disabled. Disabling forwarding of an application does not disable other support for the application. For example, if you disable forwarding of Telnet requests to helper addresses, other Telnet support on the Layer 3 Switch is not also disabled.

Enabling forwarding for a UDP application

If you want the Layer 3 Switch to forward client requests for UDP applications that the Layer 3 Switch does not forward by default, you can enable forwarding support for the port. To enable forwarding support for a UDP application, use the following method. You also can disable forwarding for an application using this method.

NOTE
You also must configure a helper address on the interface that is connected to the clients for the application. The Layer 3 Switch cannot forward the requests unless you configure the helper address. Refer to "Configuring an IP helper address" on page 66.

To enable the forwarding of SNMP trap broadcasts, enter the following command.

```
Brocade(config)# ip forward-protocol udp ntp
```

Syntax: [**no**] **ip forward-protocol udp** *udp-port-name* | *udp-port-num*

The *udp-port-name* parameter can have one of the following values. For reference, the corresponding port numbers from RFC 1340 are shown in parentheses. If you specify an application name, enter the name only, not the parentheses or the port number shown here:

- bootpc (port 68)
- bootps (port 67)
- discard (port 9)
- dns (port 53)
- dnsix (port 90)
- echo (port 7)
- mobile-ip (port 434)
- netbios-dgm (port 138)
- netbios-ns (port 137)
- ntp (port 123)
- tacacs (port 65)
- talk (port 517)
- time (port 37)
- tftp (port 69)

In addition, you can specify any UDP application by using the application UDP port number.

The *udp-port-num* parameter specifies the UDP application port number. If the application you want to enable is not listed above, enter the application port number. You also can list the port number for any of the applications listed above.

To disable forwarding for an application, enter a command such as the following.

```
Brocade(config)# no ip forward-protocol udp ntp
```

This command disables forwarding of SNMP requests to the helper addresses configured on Layer 3 Switch interfaces.

Configuring an IP helper address

To forward a client broadcast request for a UDP application when the client and server are on different networks, you must configure a helper address on the interface connected to the client. Specify the server IP address or the subnet directed broadcast address of the IP subnet the server is in as the helper address.

You can configure up to 16 helper addresses on each interface. You can configure a helper address on an Ethernet port or a virtual interface.

To configure a helper address on an interface 2 on chassis module 1, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/2
Brocade(config-if-e10000-1/1/2)# ip helper-address 1 192.168.7.6
```

The commands in this example change the CLI to the configuration level for port 1/1/2, then add a helper address for server 192.168.7.6 to the port. If the port receives a client request for any of the applications that the Layer 3 Switch is enabled to forward, the Layer 3 Switch forwards the client request to the server.

Syntax: **ip helper-address** *num ip-addr*

The *num* parameter specifies the helper address number and can be from 1 through 16.

The *ip-addr* command specifies the server IP address or the subnet directed broadcast address of the IP subnet the server is in.

BootP and DHCP relay parameter configuration

A host on an IP network can use BootP or DHCP to obtain its IP address from a BootP/DHCP server. To obtain the address, the client sends a BootP or DHCP request. The request is a subnet directed broadcast and is addressed to UDP port 67. A limited IP broadcast is addressed to IP address 255.255.255.255 and is not forwarded by the Brocade Layer 3 Switch or other IP routers.

When the BootP or DHCP client and server are on the same network, the server receives the broadcast request and replies to the client. However, when the client and server are on different networks, the server does not receive the client request, because the Layer 3 Switch does not forward the request.

You can configure the Layer 3 Switch to forward BootP/DHCP requests. To do so, configure a helper address on the interface that receives the client requests, and specify the BootP/DHCP server IP address as the address you are helping the BootP/DHCP requests to reach. Instead of the server IP address, you can specify the subnet directed broadcast address of the IP subnet the server is in.

BootP and DHCP relay parameters

The following parameters control the Layer 3 Switch forwarding of BootP and DHCP requests:

- Helper address – The BootP/DHCP server IP address. You must configure the helper address on the interface that receives the BootP/DHCP requests from the client. The Layer 3 Switch cannot forward a request to the server unless you configure a helper address for the server.
- Gateway address – The Layer 3 Switch places the IP address of the interface that received the BootP/DHCP request in the request packet Gateway Address field (sometimes called the Router ID field). When the server responds to the request, the server sends the response as a unicast packet to the IP address in the Gateway Address field. (If the client and server are directly attached, the Gateway ID field is empty and the server replies to the client using a unicast or broadcast packet, depending on the server.)

  By default, the Layer 3 Switch uses the lowest-numbered IP address on the interface that receives the request as the Gateway address. You can override the default by specifying the IP address you want the Layer 3 Switch to use.

- **Hop count** – Each router that forwards a BootP/DHCP packet increments the hop count by 1. Routers also discard a forwarded BootP/DHCP request instead of forwarding the request if the hop count is greater than the maximum number of BootP/DHCP hops allows by the router. By default, a Brocade Layer 3 Switch forwards a BootP/DHCP request if its hop count is four or less, but discards the request if the hop count is greater than four. You can change the maximum number of hops the Layer 3 Switch will allow to a value from 1 through 15.

NOTE
The BootP/DHCP hop count is not the TTL parameter.

Configuring an IP helper address

The procedure for configuring a helper address for BootP/DHCP requests is the same as the procedure for configuring a helper address for other types of UDP broadcasts. Refer to "Configuring an IP helper address" on page 64.

Configuring the BOOTP and DHCP reply source address

You can configure the Brocade device so that a BOOTP/DHCP reply to a client contains the server IP address as the source address instead of the router IP address. To do so, enter the following command at the Global CONFIG level of the CLI.

```
Brocade(config)# ip helper-use-responder-ip
```

Syntax: [**no**] **ip helper-use-responder-ip**

Changing the IP address used for stamping BootP and DHCP requests

When the Layer 3 Switch forwards a BootP/DHCP request, the Layer 3 Switch "stamps" the Gateway Address field. The default value the Layer 3 Switch uses to stamp the packet is the lowest-numbered IP address configured on the interface that received the request. If you want the Layer 3 Switch to use a different IP address to stamp requests received on the interface, use either of the following methods to specify the address.

The BootP/DHCP stamp address is an interface parameter. Change the parameter on the interface that is connected to the BootP/DHCP client.

To change the IP address used for stamping BootP/DHCP requests received on interface 1/1/1, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)# ip bootp-gateway 192.168.22.26
```

These commands change the CLI to the configuration level for port 1/1/1, then change the BootP/DHCP stamp address for requests received on port 1/1/1 to 192.168.22.26. The Layer 3 Switch will place this IP address in the Gateway Address field of BootP/DHCP requests that the Layer 3 Switch receives on port 1/1/1 and forwards to the BootP/DHCP server.

Syntax: **ip bootp-gateway** *ip-addr*

Changing the maximum number of hops to a BootP relay server

Each BootP or DHCP request includes a field Hop Count field. The Hop Count field indicates how many routers the request has passed through. When the Layer 3 Switch receives a BootP/DHCP request, the Layer 3 Switch looks at the value in the Hop Count field:

- If the hop count value is equal to or less than the maximum hop count the Layer 3 Switch allows, the Layer 3 Switch increments the hop count by one and forwards the request.
- If the hop count is greater than the maximum hop count the Layer 3 Switch allows, the Layer 3 Switch discards the request.

To change the maximum number of hops the Layer 3 Switch allows for forwarded BootP/DHCP requests, use either of the following methods.

NOTE
The BootP and DHCP hop count is not the TTL parameter.

To modify the maximum number of BootP/DHCP hops, enter the following command.

```
Brocade(config)#bootp-relay-max-hops 10
```

This command allows the Layer 3 Switch to forward BootP/DHCP requests that have passed through ten previous hops before reaching the Layer 3 Switch. Requests that have traversed 11 hops before reaching the switch are dropped. Since the hop count value initializes at zero, the hop count value of an ingressing DHCP Request packet is the number of Layer 3 routers that the packet has already traversed.

Syntax: **bootp-relay-max-hops** *1 through 15*

DHCP Server

All Brocade ICX 6650 devices can be configured to function as DHCP Servers.

Dynamic Host Configuration Protocol (DHCP) is a computer networking protocol used by devices (DHCP clients) to obtain leased (or permanent) IP addresses. DHCP is an extension of the Bootstrap Protocol (BOOTP). The differences between DHCP and BOOTP are the address allocation and renewal process.

DHCP introduces the concept of a lease on an IP address. Refer to “How DHCP Client-Based Auto-Configuration and flash image update works” on page 82. The DHCP server can allocate an IP address for a specified amount of time, or can extend a lease for an indefinite amount of time. DHCP provides greater control of address distribution within a subnet. This feature is crucial if the subnet has more devices than available IP address. In contrast to BOOTP, which has two types of messages that can be used for leased negotiation, DHCP provides 7 types of messages. Refer to “Supported options for DHCP Servers” on page 85.

DHCP allocates temporary or permanent network IP addresses to clients. When a client requests the use of an address for a time interval, the DHCP server guarantees not to reallocate that address within the requested time and tries to return the same network address each time the client makes a request. The period of time for which a network address is allocated to a client is called a lease. The client may extend the lease through subsequent requests. When the client is done with the address, they can release the address back to the server. By asking for an indefinite lease, clients may receive a permanent assignment.

In some environments, it may be necessary to reassign network addresses due to exhaustion of the available address pool. In this case, the allocation mechanism reuses addresses with expired leases.

Configuration notes for configuring DHCP servers

- DHCP server is supported in the Layer 2 and full Layer 3 software images.
- In the event of a controlled or forced switchover, a DHCP client will request from the DHCP server the same IP address and lease assignment that it had before the switchover. After the switchover, the DHCP Server feature will be automatically re-initialized on the new active controller or management module.
- If any address from the configured DHCP pool is used, for example by the DHCP server, TFTP server, etc., you must exclude the address from the network pool. For configuration instructions, refer to “Specifying addresses to exclude from the address pool” on page 76.

DHCP option 82 support

The DHCP relay agent information option (DHCP option 82) enables a DHCP relay agent to include information about itself when forwarding client-originated DHCP packets to a DHCP server. The DHCP server uses this information to implement IP address or other parameter-assignment policies.

In a metropolitan Ethernet-access environment, the DHCP server can centrally manage IP address assignments for a large number of subscribers. If DHCP option 82 is disabled, a DHCP policy can only be applied per subnet, rather than per physical port. When DCHP option 82 is enabled, a subscriber is identified by the physical port through which it connects to the network.

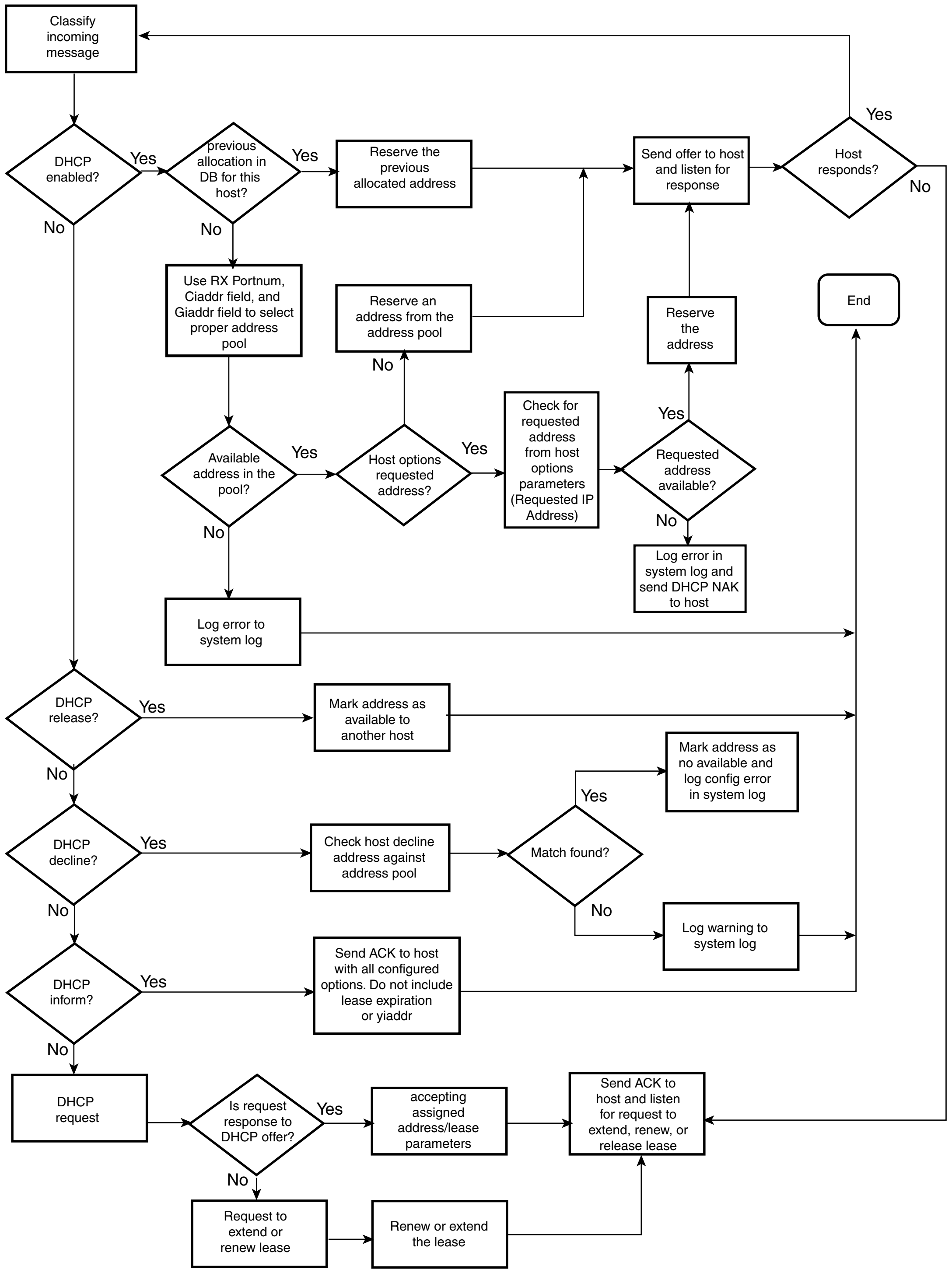
DHCP Server options

A Brocade ICX 6650 device configured as a DHCP server can support up to 1000 DHCP clients, offering them the following options:

- **NetBIOS over TCP/IP Name Server** - Specifies a list of RFC1001/1002 NBNS name servers listed in order of preference.
- **Domain Name Server** - Specifies a list of Domain Name System (RFC 1035) name servers available to the client. Servers are listed in order of preference.
- **Domain Name** - Specifies the domain name the client should use when resolving hostnames using the Domain Name system.
- **Router Option** - specifies a list of IP addresses for routers on the client subnet. Routers are listed in order of preference.
- **Subnet Mask** - Specifies the client subnet mask (per RFC950).
- **Vendor Specific Information** - Allows clients and servers to exchange vendor-specific information.
- **Boot File** - Specifies a boot image to be used by the client
- **Next Bootstrap Server** - Configures the IP address of the next server to be used for startup by the client.
- **TFTP Server** - Configures the address or name of the TFTP server available to the client.

A DHCP server assigns and manages IPv4 addresses from multiple address pools, using dynamic address allocation. The DHCP server also contains the relay agent to forward DHCP broadcast messages to network segments that do not support these types of messages.

FIGURE 7 DHCP Server configuration flow chart


Classify incoming message
DHCP enabled?
Yes
No
previous allocation in DB for this host?
Yes
No
Reserve the previous allocated address
Send offer to host and listen for response
Host responds?
Yes
No
Use RX Portnum, Ciaddr field, and Giaddr field to select proper address pool
Reserve an address from the address pool
No
Reserve the address
End
Available address in the pool?
Yes
No
Host options requested address?
Yes
Check for requested address from host options parameters (Requested IP Address)
Requested address available?
Yes
No
Log error in system log and send DHCP NAK to host
Log error to system log
DHCP release?
Yes
No
Mark address as available to another host
Mark address as no available and log config error in system log
DHCP decline?
Yes
No
Check host decline address against address pool
Match found?
Yes
No
Log warning to system log
DHCP inform?
Yes
No
Send ACK to host with all configured options. Do not include lease expiration or yiaddr
DHCP request
Is request response to DHCP offer?
Yes
No
accepting assigned address/lease parameters
Send ACK to host and listen for request to extend, renew, or release lease
Request to extend or renew lease
Renew or extend the lease

Configuring DHCP Server on a device

Perform the following steps to configure the DHCP Server feature on your device:

1. Enable DHCP Server by entering a command similar to the following.

```
Brocade(config)# ip dhcp-server enable
```

2. Create a DHCP Server address pool by entering a command similar to the following.

```
Brocade(config)# ip dhcp-server pool cabo
```

3. Configure the DHCP Server address pool by entering commands similar to the following.

```
Brocade(config-dhcp-cabo)# network 192.168.1.0/24
Brocade(config-dhcp-cabo)# domain-name brocade.com
Brocade(config-dhcp-cabo)# dns-server 192.168.1.2 192.168.1.3
Brocade(config-dhcp-cabo)# netbios-name-server 192.168.1.2
Brocade(config-dhcp-cabo)# lease 0 0 5
```

4. To disable DHCP, enter a command similar to the following.

```
Brocade(config)# no ip dhcp-server enable
```

The following sections describe the default DHCP settings, CLI commands and the options you can configure for the DHCP Server feature.

Default DHCP Server settings

Table 7 shows the default DHCP Server settings.

TABLE 7 **DHCP server default settings**

Parameter	Default Value
DHCP server	Disabled
Lease database expiration time	86400 seconds
The duration of the lease for an assigned IP address	43200 seconds (one day)
Maximum lease database expiration time	86400 seconds
DHCP server with option 82	Disabled
DHCP server unknown circuit-ID for Option 82	Permit range lookup
IP distribution mechanism	Linear

DHCP Server CLI commands

Table 8 described DHCP Server optional parameters command.

TABLE 8 **DHCP Server optional parameters command**

Command	Description
`dbexpire`	Specifies how long, in seconds, the DHCP server should wait before aborting a database transfer
`option domain-name`	Specifies the domain name for the DHCP clients.
`option domain-nameservers`	Specifies the Domain Name System (DNS) IP servers that are available to the DHCP clients.

TABLE 8 DHCP Server optional parameters command

Command	Description
`option merit-dump`	Specifies the path name of a file into which the client's core image should be placed in the event that the client crashes (the DHCP application issues an exception in case of errors such as division by zero).
`option root-path`	Specifies the name of the path that contains the client's root filesystem in NFS notation.
`option router`	Adds the default router and gateway for the DHCP clients.
`option subnet-mask`	Defines the subnet mask for the network.
`option broadcastaddress`	Defines a broadcast address for the network.
`option wins-server`	Defines the NetBIOS Windows Internet Naming Service (WINS) name servers that are available to Microsoft DHCP clients.
`option log-servers`	Defines a list of log servers available to the client.
`option bootstrapserver`	Specifies the IP address of the bootstrap server (the command fills the "siaddr" field in the DHCP packet).
`option bootstrapfilename`	Sets the name of the bootstrap file. The **no** form of this command removes the name of the bootstrap file.
`option bootfile-name`	Specifies the pathname of the boot file.
`option tftp-server`	Specifies the IP address of a TFTP server.

Table 9 describes the CLI commands that are available in the DHCP Server feature.

TABLE 9 DHCP Server CLI commands

Command	Description
ip dhcp-server arp-ping-timeout <#>	Specifies the time (in seconds) the server will wait for a response to an arp-ping packet before deleting the client from the binding database. The minimum setting is 5 seconds and the maximum time is 30 seconds. **NOTE:** Do not alter the default value unless it is necessary. Increasing the value of this timer may increase the time to get console access after a reboot.
clear ip dhcp-server binding	Deletes a specific, or all leases from the binding database. Refer to "Removing DHCP leases" on page 74.
ip dhcp-server enable	Enables the DHCP server feature. Refer to "Enabling DHCP Server" on page 74.
no ip dhcp-server mgmt	Disables DHCP server on the management port. Refer to "Disabling DHCP Server on the management port" on page 74.
ip dhcp-server pool *name*	Switches to pool configuration mode (config-dhcp-name# prompt) and creates an address pool. Refer to "Creating an address pool" on page 75.
ip dhcp-server relay-agent-echo enable	Enables relay agent echo (Option 82). Refer to "Enabling relay agent echo (Option 82)" on page 75.
ip dhcp-server *server-id*	Specifies the IP address of the selected DHCP server. Refer to "Configuring the IP address of the DHCP server" on page 75.
show ip dhcp-server binding [*address*]	Displays a specific lease entry, or all lease entries. Refer to "Displaying active lease entries" on page 78.

TABLE 9 **DHCP Server CLI commands (Continued)**

Command	Description
show ip dhcp-server address-pool *name*	Displays a specific address pool or all address pools. Refer to "Displaying address-pool information" on page 78.
show ip dhcp-server flash	Displays the lease binding database that is stored in flash memory. Refer to "Displaying lease-binding information in flash memory" on page 79.
show ip dhcp-server summary	Displays a summary of active leases, deployed address pools, undeployed address pools, and server uptime."Displaying summary DHCP server information" on page 80.
bootfile *name*	Specifies a boot image to be used by the client. Refer to "Configuring the boot image" on page 75.
deploy	Deploys an address pool configuration to the server. Refer to "Deploying an address pool configuration to the server" on page 76.
dhcp-default-router *addresses*	Specifies the IP address of the default router or routers for a client. Refer to "Specifying default routers available to the client" on page 76.
dns-server *addresses*	Specifies the IP addresses of a DNS server or servers available to the client. Refer to "Specifying DNS servers available to the client" on page 76.
domain-name *domain*	Configures the domain name for the client. Refer to "Configuring the domain name for the client" on page 76.
lease *dayshoursminutes*	Specifies the lease duration for an address pool. The default is a one-day lease. Refer to"Configuring the lease duration for the address pool" on page 76.
excluded-address [*address* |*address-low* | *address-high*]	Specifies an address or range of addresses to be excluded from the address pool. Refer to"Specifying addresses to exclude from the address pool" on page 76.
netbios-name-server *address* [*address2* | *address3*]	Specifies the IP address of a NetBIOS WINS server or servers that are available to Microsoft DHCP clients. Refer to "Configuring the NetBIOS server for DHCP clients" on page 77.
network *subnet/mask*	Configures the subnet network and mask of the DHCP address pool. Refer to "Configuring the subnet and mask of a DHCP address pool" on page 77.
next-bootstrap-server *address*	Configures the IP address of the next server to be used for startup by the client. Refer to "Configuring a next-bootstrap server" on page 77.
tftp-server *address* | name *name*	Configures the address or name of the TFTP server available to the client. Refer to "Configuring the TFTP server" on page 77.
vendor-class [*ascii* | *ip* | *hex*] *value*	Specifies the vendor type and configuration value for the DHCP client. Refer to "Configuring a vendor type and configuration value for a DHCP client" on page 77.
default-lease-time	Specifies the duration of the lease for an IP address that is assigned from a DHCP server to a DHCP client.
database tftp	Defines the TFTP IP address server for storing the DHCP database, the name of the stored file and the time period at which the stored database is synchronized with the database on the device.
database ftp	Defines the FTP IP address server for storing the DHCP database, the name of the stored file and the time period at which the stored database is synchronized with the database on the device.
max-lease-time	Specifies the maximal duration of the leases in seconds.

Removing DHCP leases

The **clear ip dhcp-server binding** command can be used to delete a specific lease, or all lease entries from the lease binding database.

```
Brocade(config)# clear ip dhcp-server binding *
```

Syntax: **clear ip dhcp-server binding** [*address* | *]

- *address* - The IP address to be deleted
- * - Clears all IP addresses

Enabling DHCP Server

The **ip dhcp-server enable** command enables DHCP Server, which is disabled by default.

Syntax: [**no**] **ip dhcp-server enable**

The **no** version of this command disables DHCP Server.

Disabling DHCP Server on the management port

By default, when DHCP Server is enabled, it responds to DHCP client requests received on the management port. If desired, you can prevent the response to DHCP client requests received on the management port, by disabling DHCP Server support on the port. When disabled, DHCP client requests that are received on the management port are silently discarded.

To disable DHCP Server on the management port, enter the following command at the global configuration level of the CLI.

```
Brocade(config)# no ip dhcp-server mgmt
```

To re-enable DHCP Server on the management port after it has been disabled, enter the **ip dhcp-server mgmt** command:

```
Brocade(config)# ip dhcp-server mgmt
```

Syntax: **[no] ip dhcp-server mgmt**

Setting the wait time for ARP-ping response

At startup, the server reconciles the lease-binding database by sending an ARP-ping packet out to every client. If there is no response to the ARP-ping packet within a set amount of time (set in seconds), the server deletes the client from the lease-binding database. The minimum setting is 5 seconds and the maximum is 30 seconds.

Syntax: **ip dhcp-server arp-ping-timeout** *num*

- *num* - The number of seconds to wait for a response to an ARP-ping packet.

NOTE
Do not alter the default value unless it is necessary. Increasing the value of this timer may increase the time to get console access after a reboot.

Creating an address pool

The **ip dhcp-server pool** command puts you in pool configuration mode, and allows you to create an address pool.

```
Brocade(config)# ip dhcp-server pool
Brocade(config-dhcp-name)# ip dhcp-server pool monterey
Brocade(config-dhcp-monterey)#
```

These commands create an address pool named monterey.

Syntax: **ip dhcp-server pool** *name*

Configuration notes for creating an address pool

- If the DHCP server address is part of a configured DHCP address pool, you must exclude the DHCP server address from the network pool. Refer to "Specifying addresses to exclude from the address pool" on page 76.
- While in DHCP server pool configuration mode, the system will place the DHCP server pool in *pending* mode and the DHCP server will not use the address pool to distribute information to clients. To activate the pool, use the **deploy** command. Refer to "Deploying an address pool configuration to the server" on page 76.

Enabling relay agent echo (Option 82)

The **ip dhcp-server relay-agent-echo enable** command activates DHCP Option 82, and enables the DHCP server to echo relay agent information in all replies.

```
Brocade(config)# ip dhcp-server relay-agent-echo enable
```

Syntax: **ip dhcp-server relay-agent-echo enable**

Configuring the IP address of the DHCP server

The **ip dhcp-server** command specifies the IP address of the selected DHCP server, as shown in this example:

```
Brocade(config)# ip dhcp-server 192.168.1.144
```

Syntax: **ip dhcp-server** *server-identifier*

- *server-identifier* - The IP address of the DHCP server

This command assigns an IP address to the selected DHCP server.

Configuring the boot image

The **bootfile** command specifies a boot image name to be used by the DHCP client.

```
Brocade(config-dhcp-cabo)# bootfile foxhound
```

In this example, the DHCP client should use the boot image called "foxhound".

Syntax: **bootfile** *name*

Deploying an address pool configuration to the server

The **deploy** command sends an address pool configuration to the DHCP server.

```
Brocade(config-dhcp-cabo)# deploy
```

Syntax: deploy

Specifying default routers available to the client

The **dhcp-default-router** command specifies the ip addresses of the default routers for a client.

Syntax: dhcp-default-router *address* [*address*, *address*]

Specifying DNS servers available to the client

The **dns-server** command specifies DNS servers that are available to DHCP clients.

```
Brocade(config-dhcp-cabo)# dns-server 192.168.1.143, 192.168.2.142
```

Syntax: dns-server *address* [*address*. *address*]

Configuring the domain name for the client

The **domain-name** command configures the domain name for the client.

```
Brocade(config-dhcp-cabo)# domain-name sierra
```

Syntax: domain-name *domain*

Configuring the lease duration for the address pool

The **lease** command specifies the lease duration for the address pool. The default is a one-day lease.

```
Brocade(config-dhcp-cabo)# lease 1 4 32
```

In this example, the lease duration has been set to one day, four hours, and 32 minutes. You can set a lease duration for just days, just hours, or just minutes, or any combination of the three.

Syntax: lease *days hours minutes*

Specifying addresses to exclude from the address pool

The **excluded-address** command specifies either a single address, or a range of addresses that are to be excluded from the address pool.

```
Brocade(config-dhcp-cabo)# excluded-address 192.168.3.44
```

Syntax: excluded-address [*address* | *address-low address-high*]

- *address* - Specifies a single address
- *address-low address-high* - Specifies a range of addresses

Configuring the NetBIOS server for DHCP clients

The **netbios-name-server** command specifies the IP address of a NetBIOS WINS server or servers that are available to Microsoft DHCP clients.

```
Brocade(config-dhcp-cabo)# netbios-name-server 192.168.1.55
```

Syntax: **netbios-name-server** *address* [*address2*, *address3*]

Configuring the subnet and mask of a DHCP address pool

This **network** command configures the subnet network and mask of the DHCP address pool.

```
Brocade(config-dhcp-cabo)# network 192.168.3.44/24
```

Syntax: **network** *subnet/mask*

Configuring a next-bootstrap server

The **next-bootstrap-server** command specifies the IP address of the next server the client should use for boot up.

```
Brocade(config-dhcp-cabo)# next-bootstrap-server 192.168.5.44
```

Syntax: **next-bootstrap-server** *address*

Configuring the TFTP server

The **tftp-server** command specifies the address or name of the TFTP server to be used by the DHCP clients.

To configure a TFTP server by specifying its IP address, enter a command similar to the following.

```
Brocade(config-dhcp-cabo)# tftp-server 192.168.5.48
```

To configure a TFTP server by specifying its server name, enter a command similar to the following.

```
Brocade(config-dhcp-cabo)# tftp-server tftp.domain.com
```

Syntax: **tftp-server** *address* | **name** *server-name*

- *address* is the IP address of the TFTP server.
- **name** configures the TFTP server specified by *server-name*.

If DHCP options 66 (TFTP server name) and 150 (TFTP server IP address) are both configured, the DHCP client ignores option 150 and tries to resolve the TFTP server name (option 66) using DNS.

Configuring a vendor type and configuration value for a DHCP client

The **vendor-class** command specifies the vendor-type and configuration value for a DHCP client.

```
Brocade(config-dhcp-cabo)# vendor class ascii waikiki
```

Syntax: **vendor-class** [*ascii* | *ip* | *hex*] *value*

Displaying DHCP Server information

The following DHCP **show** commands can be entered from any level of the CLI.

Displaying active lease entries

The **show ip dhcp-server binding** command displays a specific active lease, or all active leases, as shown in the following example:

```
Brocade# show ip dhcp-server binding
```

The following output is displayed:

```
Brocade# show ip dhcp-server binding
Bindings from all pools:
        IP Address    Client-ID/        Lease expiration Type
                      Hardware address

       192.168.1.2    0000.005d.a440     0d:0h:29m:31s   Automatic
       192.168.1.3    0000.00e1.26c0     0d:0h:29m:38s   Automatic
```

Syntax: **show ip dhcp-server binding** [*address*]

- *address* - Displays entries for this address only

Table 10 describes this output.

TABLE 10 CLI display of **show ip dhcp-server binding** command

Field	Description
IP address	The IP addresses currently in the binding database
Client ID/Hardware address	The hardware address for the client
Lease expiration	The time when this lease will expire
Type	The type of lease

Displaying address-pool information

This **show ip dhcp-server address-pool** command displays information about a specific address pool, or for all address pools.

```
Brocade# show ip dhcp-server address-pools
```

Output similar to the following is displayed, as shown here.

```
Showing all address pool(s):

Pool Name:  one
Time elapsed since last save:  0d:0h:6m:52s
Total number of active leases:  2
Address Pool State:  active
IP Address Exclusions:  192.168.1.45
IP Address Exclusions:  192.168.1.99 192.168.1.103
Pool Configured Options:
bootfile: example.bin
          dhcp-default-router:  192.168.1.1
                   dns-server:  192.168.1.100
                  domain-name:  example.com
```

```
                        lease:  0 0 30
          netbios-name-server:  192.168.1.101
                      network:  192.168.1.0 255.255.255.0
        next-bootstrap-server:  192.168.1.102
                  tftp-server:  192.168.1.103
```

Syntax: **show ip dhcp-server address-pool**[**s**] [*name*]

- **address-pool**[**s**] - If you enter address-pools, the display will show all address pools
- *name* - Displays information about a specific address pool

Table 11 describes this output.

TABLE 11 CLI display of **show ip dhcp-server address pools** command

Field	Description
Pool name	The name of the address pool
Time elapsed since last save	The time that has elapsed since the last save.
Total number of active leases	The number of leases that are currently active.
Address pool state	The state of the address pool (active or inactive).
IP Address exclusions	IP addresses that are not included in the address pool
Pool configured options	
bootfile	The name of the bootfile
dhcp-server-router	The address of the DHCP server router
dns-server	The address of the dns server
domain-name	The name of the domain
lease	The identifier for the lease
netbios-name server	The address of the netbios name server
network	The address of the network
next-bootstrap-server	The address of the next-bootstrap server
tftp-server	The address of the TFTP server

Displaying lease-binding information in flash memory

The **show ip dhcp-server flash** command displays the lease-binding database that is stored in flash memory.

```
Brocade# show ip dhcp-server flash
```

The following information is displayed.

```
Brocade# show ip dhcp-server flash
Address Pool Binding:
        IP Address    Client-ID/        Lease expiration Type
                      Hardware address

       192.168.1.2    0000.005d.a440     0d:0h:18m:59s   Automatic
       192.168.1.3    0000.00e1.26c0      0d:0h:19m:8s   Automatic
```

Syntax: **show ip dhcp-server flash**

Table 12 describes this output.

TABLE 12 CLI display of show ip dhcp-server flash command

Field	Description
IP address	The IP address of the flash memory lease-binding database
Client-ID/Hardware address	The address of the client
Lease expiration	The time when the lease will expire
Type	The type of lease

Displaying summary DHCP server information

The **show ip dhcp-server summary** command displays information about active leases, deployed address-pools, undeployed address-pools, and server uptime.

```
Brocade# show ip dhcp-server summary
```

The following information is displayed.

```
DHCP Server Summary:

                    Total number of active leases:  2
           Total number of deployed address-pools:  1
         Total number of undeployed address-pools:  0
                                    Server uptime: 0d:0h:8m:27s
```

Syntax: **show ip dhcp-server summary**

Table describes this output.

CLI display of show ip dhcp-server summary command

Field	Description
Total number of active leases	Indicates the number of leases that are currently active
Total number of deployed address-pools	The number of address pools currently in use.
Total number of undeployed address-pools	The number of address-pools being held in reserve.
Server uptime	The amount of time that the server has been active.

DHCP Client-Based Auto-Configuration and flash image update

DHCP Client-Based Auto-Configuration allows Layer 2 and Layer 3 devices to automatically obtain leased IP addresses through a DHCP server, negotiate address lease renewal, and obtain flash image and configuration files.

DHCP Client-Based Auto-Configuration occurs as follows.

1. The IP address validation and lease negotiation enables the DHCP client (a Brocade Layer 2 or Layer 3 device) to automatically obtain and configure an IP address, as follows:

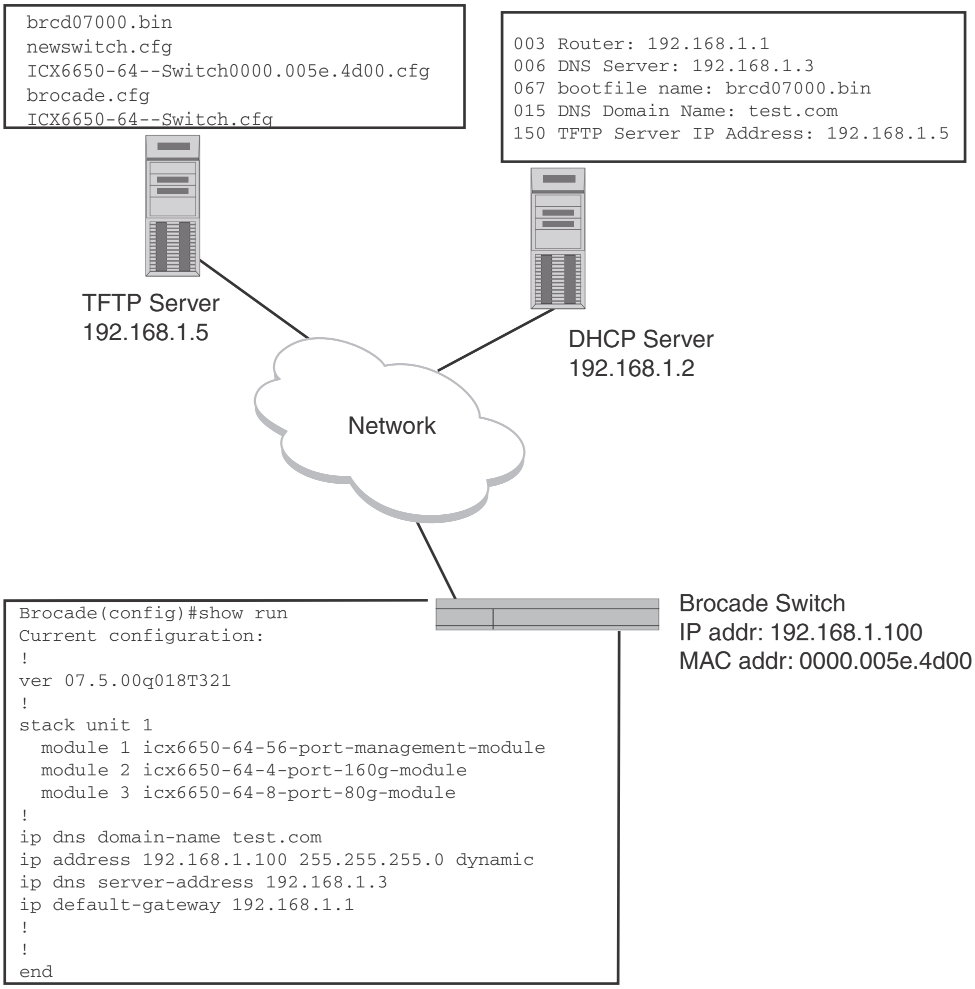
    - One lease is granted for each Layer 2 device. if the device is configured with a static IP address, the DHCP Auto-Configuration feature is automatically disabled.

- For a Layer 3 device, one leased address is granted (per device) to the interface that first receives a response from the DHCP server.

2. If **auto-update** is enabled, the TFTP flash image is downloaded and updated. The device compares the filename of the requested flash image with the image stored in flash. If the filenames are different, then the device will download the new image from a TFTP server, write the downloaded image to flash, then reload the device.
3. In the final step, TFTP configuration download and update, the device downloads a configuration file from a TFTP server and saves it as the running configuration.

Figure 8 shows how DHCP Client-Based Auto Configuration works.

FIGURE 8 **DHCP Client-Based Auto-Configuration**



```
brcd07000.bin
newswitch.cfg
ICX6650-64--Switch0000.005e.4d00.cfg
brocade.cfg
ICX6650-64--Switch.cfg
```

```
003 Router: 192.168.1.1
006 DNS Server: 192.168.1.3
067 bootfile name: brcd07000.bin
015 DNS Domain Name: test.com
150 TFTP Server IP Address: 192.168.1.5
```

```
Brocade(config)#show run
Current configuration:
!
ver 07.5.00q018T321
!
stack unit 1
  module 1 icx6650-64-56-port-management-module
  module 2 icx6650-64-4-port-160g-module
  module 3 icx6650-64-8-port-80g-module
!
ip dns domain-name test.com
ip address 192.168.1.100 255.255.255.0 dynamic
ip dns server-address 192.168.1.3
ip default-gateway 192.168.1.1
!
!
end
```

Configuration notes and feature limitations for DHCP Cient-Based Auto-Configuration

- For Layer 3 devices, this feature is available for the default VLAN only. For Layer 2 devices, this feature is available for default VLANs and management VLANs. This feature is not supported on virtual interfaces (VEs), trunked ports, or LACP ports.

- Although the DHCP server may provide multiple addresses, only one IP address is installed at a time.
- This feature is not supported together with DHCP snooping.

The following configuration rules apply to flash image update:

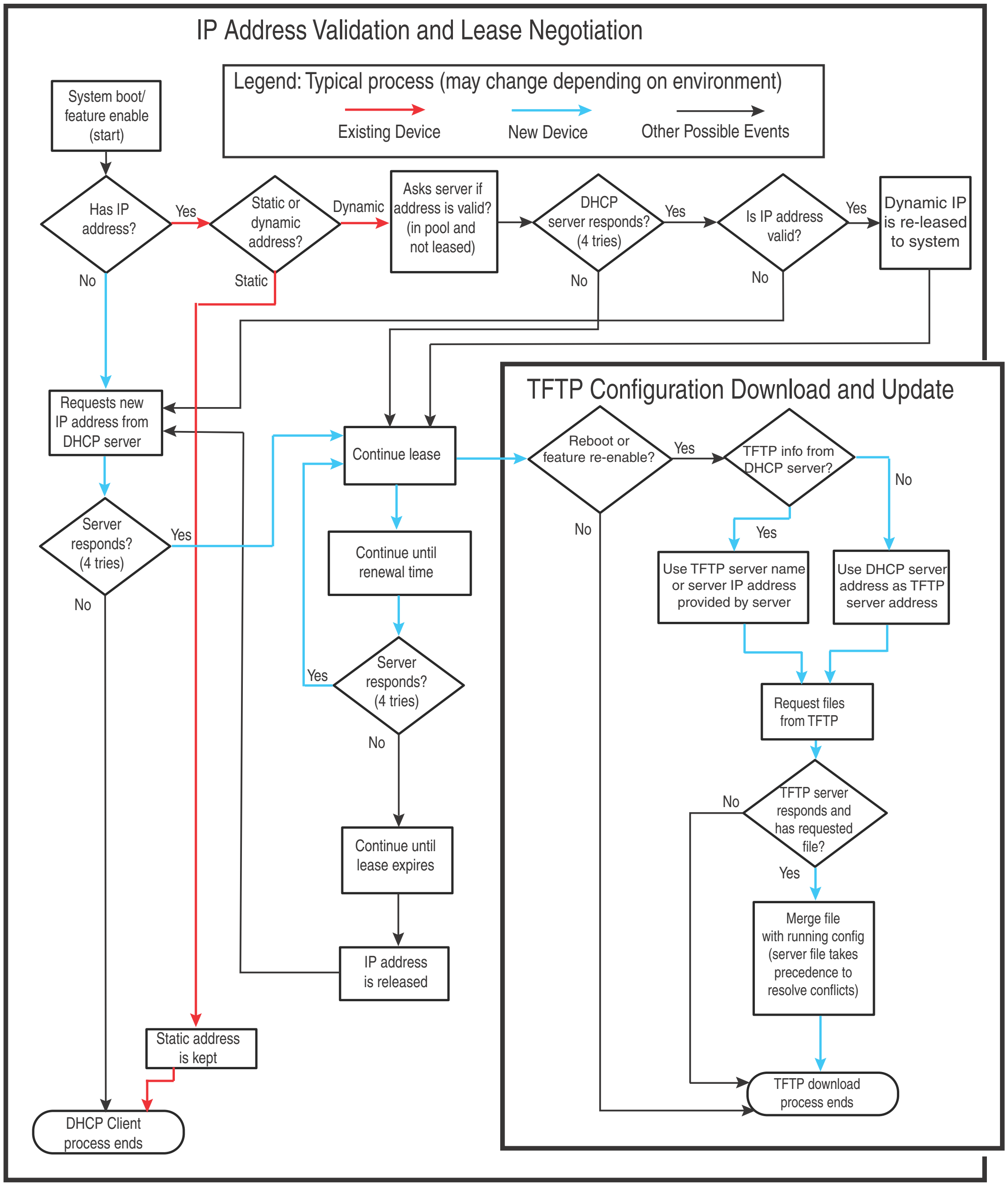
- To enable flash image update (**ip dhcp-client auto-update enable** command), also enable auto-configuration (**ip dhcp-client enable** command).
- The image filename to be updated must have the extension **.bin**.
- The DHCP option 067 bootfile name will be used for image update if it has the extension **.bin**.
- The DHCP option 067 bootfile name will be used for configuration download if it does not have the extension **.bin**.
- If the DHCP option 067 bootfile name is not configured or does not have the extension **.bin**, then the auto-update image will not occur.

How DHCP Client-Based Auto-Configuration and flash image update works

Auto-Configuration and Auto-update are enabled by default. To disable this feature, refer to "Disabling or re-enabling Auto-Configuration" on page 86 and "Disabling or re-enabling Auto-Update" on page 86, respectively.

The steps of the Auto-Configuration and Auto-update process are described in Figure 9, and in the description that follows the flowchart.

FIGURE 9 The DHCP Client-Based Auto-Configuration steps


IP Address Validation and Lease Negotiation
Legend: Typical process (may change depending on environment)
Existing Device
New Device
Other Possible Events
System boot/ feature enable (start)
Has IP address?
Yes
No
Static or dynamic address?
Dynamic
Static
Asks server if address is valid? (in pool and not leased)
DHCP server responds? (4 tries)
Yes
No
Is IP address valid?
Yes
No
Dynamic IP is re-leased to system
Requests new IP address from DHCP server
Server responds? (4 tries)
Yes
No
Continue lease
Continue until renewal time
Server responds? (4 tries)
Yes
No
Continue until lease expires
IP address is released
Static address is kept
DHCP Client process ends
TFTP Configuration Download and Update
Reboot or feature re-enable?
Yes
No
TFTP info from DHCP server?
Yes
No
Use TFTP server name or server IP address provided by server
Use DHCP server address as TFTP server address
Request files from TFTP
TFTP server responds and has requested file?
No
Yes
Merge file with running config (server file takes precedence to resolve conflicts)
TFTP download process ends

Validate the IP address and lease negotiation

1. At boot-up, the device automatically checks its configuration for an IP address.
2. If the device *does not have* a static IP address, it requests the lease of an address from the DHCP server:
   - If the server responds, it leases an IP address to the device for the specified lease period.
   - If the server does not respond (after four tries) the DHCP Client process is ended.

3. If the device ***has*** a ***dynamic*** address, the device asks the DHCP server to validate that address. If the server does not respond, the device will continue to use the existing address until the lease expires. If the server responds, and the IP address is outside of the DHCP address pool or has been leased to another device, it is automatically rejected, and the device receives a new IP address from the server. If the existing address is valid, the lease continues.

   NOTE
   The lease time interval is configured on the DHCP server, not on the client device. The **ip dhcp-client lease** command is set by the system, and is non-operational to a user.

4. If the existing address is **static**, the device keeps it and the DHCP Client process is ended.
5. For a leased IP address, when the lease interval reaches the renewal point, the device requests a renewal from the DHCP server:
   - If the device is ***able*** to contact the DHCP server at the renewal point in the lease, the DHCP server extends the lease. This process can continue indefinitely.
   - If the device is ***unable*** to reach the DHCP server after four attempts, it continues to use the existing IP address until the lease expires. When the lease expires, the dynamic IP address is removed and the device contacts the DHCP server for a new address. If the device is still unable to contact the DHCP server after four attempts, the process is ended.

The TFTP Flash image download and update step

NOTE
This process only occurs when the client device reboots, or when DHCP-client has been disabled and then re-enabled.

Once a lease is obtained from the server (described in “Validate the IP address and lease negotiation” on page 83), the device compares the filename of the requested flash image with the image stored in flash.

- If the .bin filenames match, then the DHCP client skips the flash image download. If **auto-configuration** is enabled, the DHCP client proceeds with downloading the configuration files as described in “The TFTP configuration download and update step”.
- If the .bin filenames are different, then the DHCP client downloads the new image from a TFTP server, then writes the downloaded image to flash.

  The code determines which flash (i.e., primary or secondary) to use based on how the device is booted. Once the flash is updated with the newer flash image, the device is reloaded If **auto-configuration** is enabled, the DHCP client then proceeds to download the configuration files described in “The TFTP configuration download and update step”.

The TFTP configuration download and update step

NOTE
This process only occurs when the client device reboots, or when Auto-Configuration has been disabled and then re-enabled.

1. When the device reboots, or the Auto-Configuration feature has been disabled and then re-enabled, the device uses information from the DHCP server to contact the TFTP server to update the running-configuration file:
    - If the DHCP server provides a TFTP server name or IP address, the device uses this information to request files from the TFTP server.
    - If the DHCP server does not provide a TFTP server name or IP address, the device requests the configuration files from the DHCP server.
2. The device requests the configuration files from the TFTP server by asking for filenames in the following order:
    - bootfile name provided by the DHCP server (if configured)
    - hostnameMAC-config.cfg, for example:

      ```
      ICX6650-64-Router0000.008f.23b7-config.cfg
      ```
    - hostnameMAC.cfg, for example:

      ```
      ICX6650-64-Router0000.008f.23b7.cfg
      ```
    - brocade.cfg (applies to all devices), for example:

      ```
      brocade.cfg
      ```
    - <icx6650> -<switch | router>.cfg (applies to Layer 2 or base Layer 3 devices), for example:

      ```
      icx6650-router.cfg (Brcd6650 Layer 2)
      icx6650-router.cfg (Brcd6650 Layer 3)
      ```

    If the device is successful in contacting the TFTP server and the server has the configuration file, the files are merged. If there is a conflict, the server file takes precedence.

    If the device is *unable* to contact the TFTP server or if the files are not found on the server, the TFTP part of the configuration download process ends.

Supported options for DHCP Servers

DHCP Client supports the following options:

- 001 - subnetmask
- 003 - router ip
- 015 - domain name
- 006 - domain name server
- 012 - hostname (optional)
- 066 - TFTP server name (only used for Client-Based Auto Configuration)
- 067 - bootfile name
- 150 - TFTP server IP address (private option, datatype = IP Address)

Configuration notes for DHCP servers

- When using DHCP on a router, if you have a DHCP address for one interface, and you want to connect to the DHCP server from another interface, you must disable DHCP on the first interface, then enable DHCP on the second interface.

- When DHCP is disabled, and then re-enabled, or if the system is rebooted, the TFTP process requires approximately three minutes to run in the background before file images can be downloaded manually.
- Once a port is assigned a leased IP address, it is bound by the terms of the lease regardless of the link state of the port.

Disabling or re-enabling Auto-Configuration

For a switch, you can disable or enable this feature using the following commands.

```
Brocade(config)# ip dhcp-client enable
Brocade(config)# no ip dhcp-client enable
```

For a router, you can disable or enable this feature using the following commands.

```
Brocade(config-if-e10000-1/1/1)# ip dhcp-client enable
Brocade(config-if-e10000-1/1/1)# no ip dhcp-client enable
```

Syntax: [**no**] **ip dhcp-client enable**

Disabling or re-enabling Auto-Update

Auto-update is enabled by default. To disable it, use the following command.

```
Brocade(config)# no ip dhcp-client auto-update enabled
```

To re-enable auto-update after it has been disabled, use the following command.

```
Brocade(config)# ip dhcp-client auto-update enabled
```

Syntax: [**no**] **ip dhcp-client auto-update enabled**

Displaying DHCP configuration information

The following example shows output from the **show ip** command for Layer 2 devices.

```
Brocade(config)# show ip

     Switch IP address: 10.44.16.116

           Subnet mask: 255.255.255.0

Default router address: 10.44.16.1
   TFTP server address: 10.44.16.41
Configuration filename: brocade.cfg
        Image filename: None
```

The following example shows output from the **show ip address** command for a Layer 2 device.

```
Brocade(config)# show ip address
  IP Address       Type      Lease Time       Interface
10.44.16.116       Dynamic    174              1/1/1
```

The following example shows output from the **show ip address** command for a Layer 3 device.

```
Brocade(config)# show ip address
  IP Address       Type      Lease Time       Interface
 10.44.3.233       Dynamic   672651           1/1/2
  10.0.0.1          Static    N/A              1/1/5
```

The following example shows a Layer 2 device configuration as a result of the **show run** command.

```
Brocade(config)# show run
Current configuration:
!
ver 07.5.00b1T323
!
stack unit 1
  module 1 icx6650-56-port-management-module
  module 2 icx6650-4-port-40g-module
  module 3 icx6650-8-port-10g-module
!
!
ip address 10.44.16.116 255.255.255.0 dynamic
ip dns server-address 10.44.16.41
ip dhcp-client lease 174
ip default-gateway 10.44.16.1
```

The following example shows a Layer 3 device configuration as a result of the **show run** command.

```
Brocade(config)# show run
Current configuration:
!
ver 07.5.00b1T323
!
stack unit 1
  module 1 icx6650-56-port-management-module
  module 2 icx6650-4-port-40g-module
  module 3 icx6650-8-port-10g-module
!
vlan 1 name DEFAULT-VLAN by port
!
ip dns domain-name test.com
ip dns server-address 10.44.3.111
interface ethernet 1/1/2
 ip address 10.44.3.233 255.255.255.0 dynamic
 ip dhcp-client lease 691109
!
interface ethernet 1/1/5
 ip address 10.0.0.1 255.0.0.0
 ip helper-address 1 10.44.3.111
!
end
```

DHCP Log messages

The following DHCP notification messages are sent to the log file.

```
2d01h48m21s:I: DHCPC: existing ip address found, no further action needed by DHCPC
2d01h48m21s:I: DHCPC: Starting DHCP Client service
2d01h48m21s:I: DHCPC: Stopped DHCP Client service
2d01h48m21s:I: DHCPC: ICX6650 Switch running-configuration changed
2d01h48m21s:I: DHCPC: sending TFTP request for bootfile name icx6650-switch.cfg
2d01h48m21s:I: DHCPC: TFTP unable to download running-configuration
2d01h48m21s:I: DHCPC: Found static IP Address 192.168.1.1 subnet mask
255.255.255.0 on port 1/1/5
2d01h48m21s:I: DHCPC: Client service found no DHCP server(s) on 3 possible subnet
2d01h48m21s:I: DHCPC: changing 1/1/3 protocol from stopped to running
```

Configuring IP parameters – Layer 2 Switches

The following sections describe how to configure IP parameters on a Brocade Layer 2 Switch.

NOTE
This section describes how to configure IP parameters for Layer 2 Switches. For IP configuration information for Layer 3 Switches, refer to "Configuring IP parameters – Layer 3 Switches" on page 19.

Configuring the management IP address and specifying the default gateway

To manage a Layer 2 Switch using Telnet or Secure Shell (SSH) CLI connections, you must configure an IP address for the Layer 2 Switch. Optionally, you also can specify the default gateway.

Brocade devices support both classical IP network masks (Class A, B, and C subnet masks, and so on) and Classless Interdomain Routing (CIDR) network prefix masks:

- To enter a classical network mask, enter the mask in IP address format. For example, enter "192.168.22.99 255.255.255.0" for an IP address with a Class-C subnet mask.
- To enter a prefix network mask, enter a forward slash (/) and the number of bits in the mask immediately after the IP address. For example, enter "192.168.22.99/24" for an IP address that has a network mask with 24 significant bits (ones).

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the display to prefix format. Refer to "Changing the network mask display to prefix format" on page 113.

Assigning an IP address to a Brocade Layer 2 switch

To assign an IP address to a Brocade Layer 2 Switch, enter a command such as the following at the global CONFIG level.

```
Brocade(config)# ip address 192.168.6.110 255.255.255.0
```

Syntax: **ip address** *ip-addr ip-mask*

or

Syntax: **ip address** *ip-addr*/*mask-bits*

You also can enter the IP address and mask in CIDR format, as follows.

```
Brocade(config)# ip address 192.168.6.1/24
```

To specify the Layer 2 Switch default gateway, enter a command such as the following.

```
Brocade(config)# ip default-gateway 192.168.6.1
```

Syntax: **ip default-gateway** *ip-addr*

NOTE
When configuring an IP address on a Layer 2 switch that has multiple VLANs, make sure the configuration includes a designated management VLAN that identifies the VLAN to which the global IP address belongs. Refer to the section “Designated VLAN for Telnet management sessions to a Layer 2 Switch” in the *Brocade ICX 6650 Security Configuration Guide*.

Configuring Domain Name Server resolver

The Domain Name Server (DNS) resolver feature lets you use a host name to perform Telnet, ping, and traceroute commands. You can also define a DNS domain on a Brocade Layer 2 Switch or Layer 3 Switch and thereby recognize all hosts within that domain. After you define a domain name, the Brocade Layer 2 Switch or Layer 3 Switch automatically appends the appropriate domain to the host and forwards it to the domain name server.

For example, if the domain “newyork.com” is defined on a Brocade Layer 2 Switch or Layer 3 Switch and you want to initiate a ping to host “NYC01” on that domain, you need to reference only the host name in the command instead of the host name and its domain name. For example, you could enter either of the following commands to initiate the ping.

```
Brocade# ping nyc01
Brocade# ping nyc01.newyork.com
```

Defining a DNS entry

You can define up to four DNS servers for each DNS entry. The first entry serves as the primary default address. If a query to the primary address fails to be resolved after three attempts, the next gateway address is queried (also up to three times). This process continues for each defined gateway address until the query is resolved. The order in which the default gateway addresses are polled is the same as the order in which you enter them.

Suppose you want to define the domain name of newyork.com on a Layer 2 Switch and then define four possible default DNS gateway addresses. To do so, enter the following commands.

```
Brocade(config)# ip dns domain-name newyork.com
Brocade(config)# ip dns server-address 192.168.22.199 192.168.7.15 192.168.17.25
192.168.10.15
```

Syntax: **ip dns server-address** *ip-addr* [*ip-addr*] [*ip-addr*] [*ip-addr*]

In this example, the first IP address in the **ip dns server-address...** command becomes the primary gateway address and all others are secondary addresses. Because IP address 192.168.10.15 is the last address listed, it is also the last address consulted to resolve a query.

Using a DNS name to initiate a trace route

Suppose you want to trace the route from a Brocade Layer 2 Switch to a remote server identified as NYC02 on domain newyork.com. Because the newyork.com domain is already defined on the Layer 2 Switch, you need to enter only the host name, NYC02, as noted in the following command.

```
Brocade# traceroute nyc02
```

Syntax: **traceroute** *host-ip-addr* [**maxttl** *value*] [**minttl** *value*] [**numeric**] [**timeout** *value*]
[**source-ip** *ip addr*]

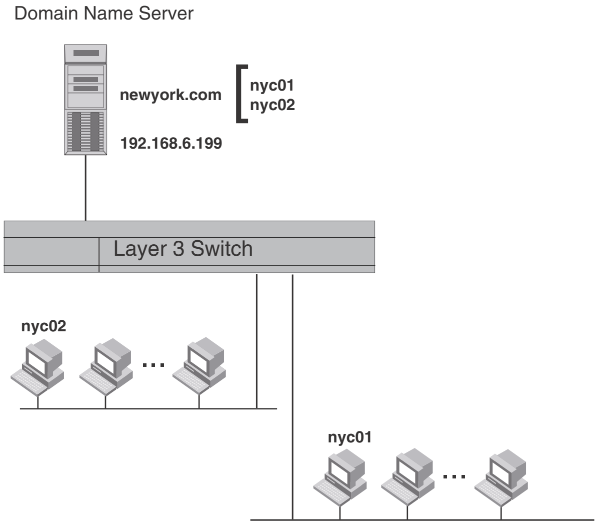
The only required parameter is the IP address of the host at the other end of the route.

After you enter the command, a message indicating that the DNS query is in process and the current gateway address (IP address of the domain name server) being queried appear on the screen.

```
Type Control-c to abort
Sending DNS Query to 192.168.22.199
Tracing Route to IP node 192.168.22.80
To ABORT Trace Route, Please use stop-traceroute command.
 Traced route to target IP node 192.168.22.80:
   IP Address          Round Trip Time1    Round Trip Time2
  192.168.6.30         93 msec             121 msec
```

NOTE
In the previous example, 192.168.22.199 is the IP address of the domain name server (default DNS gateway address), and 192.168.22.80 represents the IP address of the NYC02 host.

FIGURE 10 Querying a Host on the newyork.com Domain



Changing the TTL threshold

The time to live (TTL) threshold prevents routing loops by specifying the maximum number of router hops an IP packet originated by the Layer 2 Switch can travel through. Each device capable of forwarding IP that receives the packet decrements (decreases) the packet TTL by one. If a router receives a packet with a TTL of 1 and reduces the TTL to zero, the router drops the packet.

The default TTL is 64. You can change the TTL to a value from 1 through 255.

To modify the TTL threshold to 25, enter the following commands.

```
Brocade(config)# ip ttl 25
Brocade(config)# exit
```

Syntax: **ip ttl** *1-255*

DHCP Assist configuration

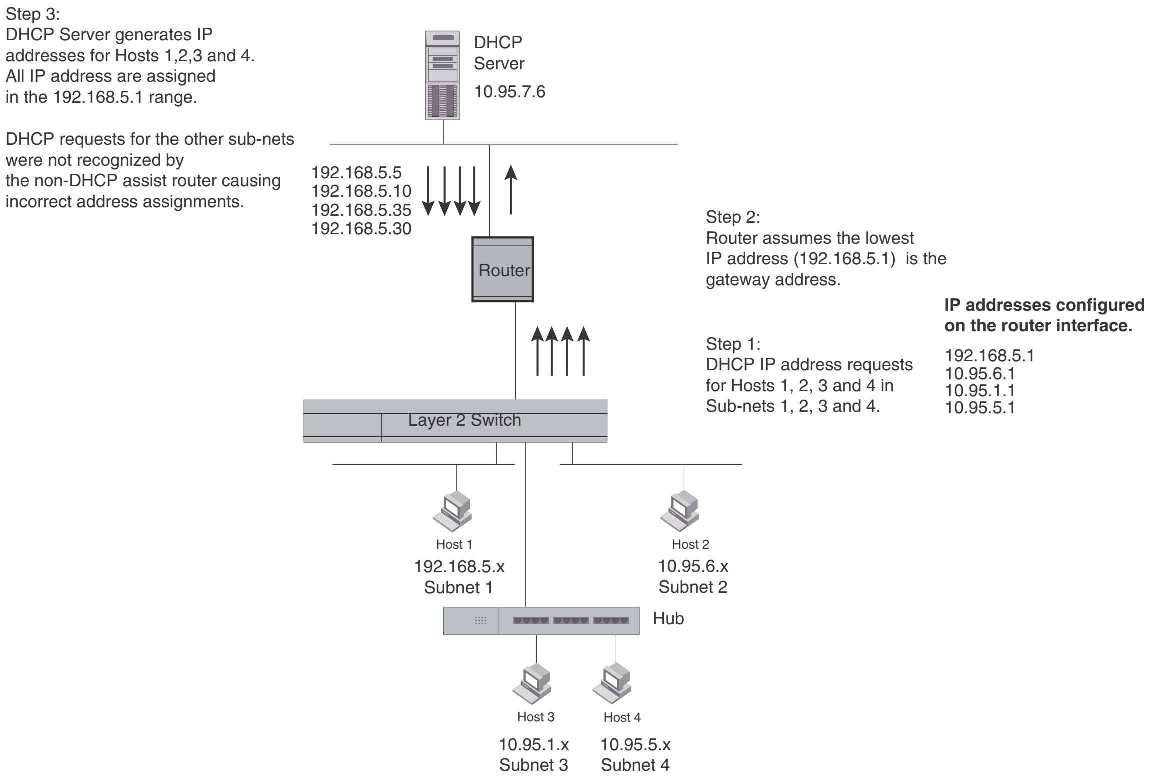
DHCP Assist allows a Brocade Layer 2 Switch to assist a router that is performing multi-netting on its interfaces as part of its DHCP relay function.

DHCP Assist ensures that a DHCP server that manages multiple IP subnets can readily recognize the requester IP subnet, even when that server is not on the client local LAN segment. The Brocade Layer 2 Switch does so by stamping each request with its IP gateway address in the DHCP discovery packet.

NOTE
Brocade Layer 3 Switches provide BootP/DHCP assistance by default on an individual port basis. Refer to "Changing the IP address used for stamping BootP and DHCP requests" on page 66.

By allowing multiple subnet DHCP requests to be sent on the same wire, you can reduce the number of router ports required to support secondary addressing as well as reduce the number of DHCP servers required, by allowing a server to manage multiple subnet address assignments.

FIGURE 11 DHCP requests in a network without DHCP Assist on the Layer 2 Switch


Step 3:
DHCP Server generates IP addresses for Hosts 1,2,3 and 4. All IP address are assigned in the 192.168.5.1 range.
DHCP requests for the other sub-nets were not recognized by the non-DHCP assist router causing incorrect address assignments.
DHCP Server
10.95.7.6
192.168.5.5
192.168.5.10
192.168.5.35
192.168.5.30
Router
Step 2:
Router assumes the lowest IP address (192.168.5.1) is the gateway address.
IP addresses configured on the router interface.
192.168.5.1
10.95.6.1
10.95.1.1
10.95.5.1
Step 1:
DHCP IP address requests for Hosts 1, 2, 3 and 4 in Sub-nets 1, 2, 3 and 4.
Layer 2 Switch
Host 1
192.168.5.x
Subnet 1
Host 2
10.95.6.x
Subnet 2
Hub
Host 3
10.95.1.x
Subnet 3
Host 4
10.95.5.x
Subnet 4

In a network operating without DHCP Assist, hosts can be assigned IP addresses from the wrong subnet range because a router with multiple subnets configured on an interface cannot distinguish among DHCP discovery packets received from different subnets.

For example, in Figure 11, a host from each of the four subnets supported on a Layer 2 Switch requests an IP address from the DHCP server. These requests are sent transparently to the router. Because the router is unable to determine the origin of each packet by subnet, it assumes the lowest IP address or the 'primary address' is the gateway for all ports on the Layer 2 Switch and stamps the request with that address.

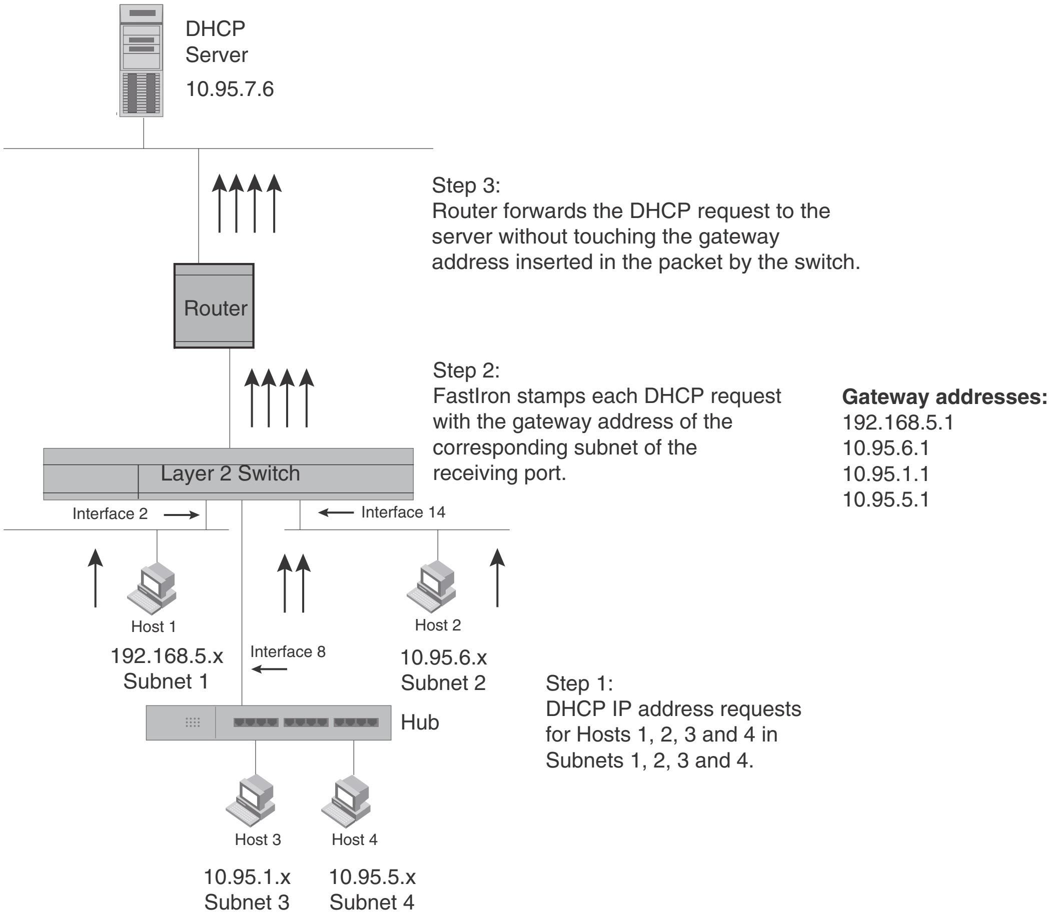
When the DHCP request is received at the server, it assigns all IP addresses within that range only.

With DHCP Assist enabled on a Brocade Layer 2 Switch, correct assignments are made because the Layer 2 Switch provides the stamping service.

How DHCP Assist works

Upon initiation of a DHCP session, the client sends out a DHCP discovery packet for an address from the DHCP server as seen in Figure 12. When the DHCP discovery packet is received at a Brocade Layer 2 Switch with the DHCP Assist feature enabled, the gateway address configured on the receiving interface is inserted into the packet. This address insertion is also referred to as stamping.

FIGURE 12 DHCP requests in a network with DHCP Assist operating on a FastIron Switch

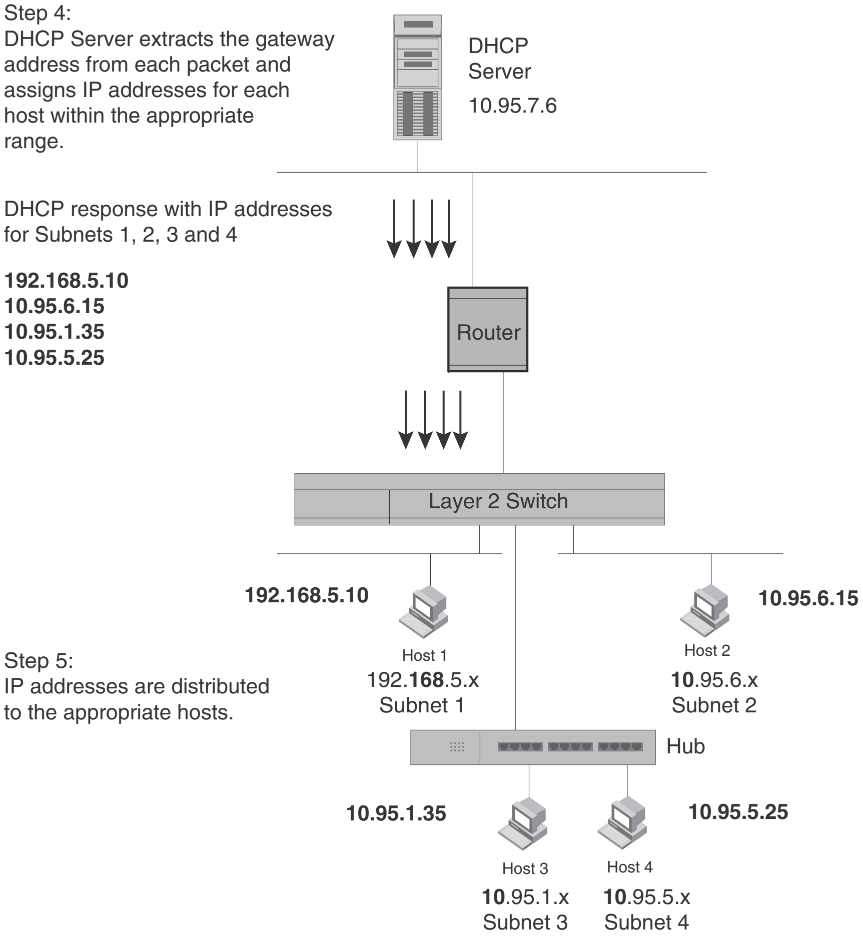

DHCP
Server
10.95.7.6
Step 3:
Router forwards the DHCP request to the server without touching the gateway address inserted in the packet by the switch.
Router
Step 2:
FastIron stamps each DHCP request with the gateway address of the corresponding subnet of the receiving port.
Gateway addresses:
192.168.5.1
10.95.6.1
10.95.1.1
10.95.5.1
Layer 2 Switch
Interface 2
Interface 14
Host 1
192.168.5.x
Subnet 1
Interface 8
Host 2
10.95.6.x
Subnet 2
Step 1:
DHCP IP address requests for Hosts 1, 2, 3 and 4 in Subnets 1, 2, 3 and 4.
Hub
Host 3
Host 4
10.95.1.x
Subnet 3
10.95.5.x
Subnet 4

When the stamped DHCP discovery packet is then received at the router, it is forwarded to the DHCP server. The DHCP server then extracts the gateway address from each request and assigns an available IP address within the corresponding IP subnet (Figure 13). The IP address is then forwarded back to the workstation that originated the request.

NOTE
When DHCP Assist is enabled on any port, Layer 2 broadcast packets are forwarded by the CPU. Unknown unicast and multicast packets are still forwarded in hardware, although selective packets such as IGMP, are sent to the CPU for analysis. When DHCP Assist is not enabled, Layer 2 broadcast packets are forwarded in hardware.

NOTE
The DHCP relay function of the connecting router must be turned on.

FIGURE 13 DHCP offers are forwarded back toward the requestors


Step 4:
DHCP Server extracts the gateway address from each packet and assigns IP addresses for each host within the appropriate range.
DHCP Server
10.95.7.6
DHCP response with IP addresses for Subnets 1, 2, 3 and 4
192.168.5.10
10.95.6.15
10.95.1.35
10.95.5.25
Router
Layer 2 Switch
192.168.5.10
Host 1
192.168.5.x
Subnet 1
10.95.6.15
Host 2
10.95.6.x
Subnet 2
Step 5:
IP addresses are distributed to the appropriate hosts.
Hub
10.95.1.35
10.95.5.25
Host 3
10.95.1.x
Subnet 3
Host 4
10.95.5.x
Subnet 4

NOTE
When DHCP Assist is enabled on any port, Layer 2 broadcast packets are forwarded by the CPU. Unknown unicast and multicast packets are still forwarded in hardware, although selective packets such as IGMP are sent to the CPU for analysis. When DHCP Assist is not enabled, Layer 2 broadcast packets are forwarded in hardware.

Configuring DHCP Assist

You can associate a gateway list with a port. You must configure a gateway list when DHCP Assist is enabled on a Brocade Layer 2 Switch. The gateway list contains a gateway address for each subnet that will be requesting addresses from a DHCP server. The list allows the stamping process to occur. Each gateway address defined on the Layer 2 Switch corresponds to an IP address of the Brocade router interface or other router involved.

Up to eight addresses can be defined for each gateway list in support of ports that are multi-homed. When multiple IP addresses are configured for a gateway list, the Layer 2 Switch inserts the addresses into the discovery packet in a round robin fashion.

Up to 32 gateway lists can be defined for each Layer 2 Switch.

Example

To create the configuration indicated in Figure 12 and Figure 13, enter commands such as the following.

```
Brocade(config)# dhcp-gateway-list 1 192.168.5.1
Brocade(config)# dhcp-gateway-list 2 10.95.6.1
Brocade(config)# dhcp-gateway-list 3 10.95.1.1 10.95.5.1
Brocade(config)# interface ethernet 1/1/2
Brocade(config-if-e10000-1/1/2)# dhcp-gateway-list 1
Brocade(config-if-e10000-1/1/2)# interface ethernet 1/1/8
Brocade(config-if-e10000-1/1/8)# dhcp-gateway-list 3
Brocade(config-if-e10000-1/1/8)# interface ethernet 1/1/14
Brocade(config-if-e10000-1/1/14)# dhcp-gateway-list 2
```

Syntax: dhcp-gateway-list *num ip-addr*

IPv4 point-to-point GRE tunnels

This section describes support for point-to-point Generic Routing Encapsulation (GRE) tunnels and how to configure them on a Brocade device.

GRE tunnels support includes, but is not limited to, the following:

- IPv4 over GRE tunnels. IPv6 over GRE tunnels is not supported.
- Static and dynamic unicast routing over GRE tunnels
- Multicast routing over GRE tunnels
- Hardware forwarding of IP data traffic across a GRE tunnel.
- Path MTU Discovery (PMTUD)

IPv4 GRE tunnel overview

Generic Routing Encapsulation is described in RFC 2784. Generally, GRE provides a way to encapsulate arbitrary packets (payload packet) inside of a transport protocol, and transmit them from one tunnel endpoint to another. The payload is encapsulated in a GRE packet. The resulting GRE packet is then encapsulated in a delivery protocol, then forwarded to the tunnel destination. At the tunnel destination, the packet is decapsulated to reveal the payload. The payload is then forwarded to its final destination.

Brocade IPv6-capable devices allow the tunneling of packets of the following protocols over an IPv4 network using GRE:

- OSPF V2
- BGP4
- RIP V1 and V2

GRE packet structure and header format

Figure 14 shows the structure of a GRE encapsulated packet.

FIGURE 14 **GRE encapsulated packet structure**

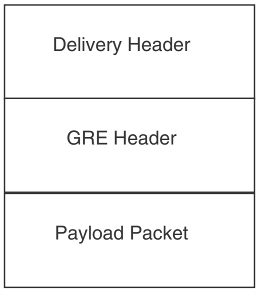
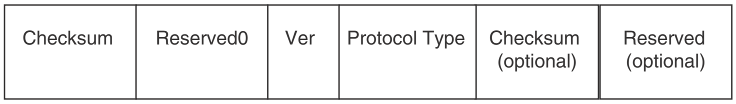


Figure 15 shows the GRE header format.

FIGURE 15 **GRE header format**



The GRE header has the following fields:

- **Checksum** – 1 bit. This field is assumed to be zero in this version. If set to 1, this means that the **Checksum (optional)** and **Reserved (optional)** fields are present and the **Checksum (optional)** field contains valid information.
- **Reserved0** – 12 bits. If bits 1 - 5 are non-zero, then a receiver must discard the packet unless RFC 1701 is implemented. Bits 6 - 12 are reserved for future use and must be set to zero in transmitted packets. This field is assumed to be zero in this version.
- **Ver** – 3 bits. The GRE protocol version. This field must be set to zero in this version.
- **Protocol Type** – 16 bits. The Ethernet protocol type of the packet, as defined in RFC 1700.
- **Checksum (optional)** – 16 bits. This field is optional. It contains the IP checksum of the GRE header and the payload packet.
- **Reserved (optional)** – 16 bits. This field is optional. It is reserved for Brocade internal use.

Path MTU Discovery (PMTUD) support

Brocade IronWare software supports the following RFCs for handling large packets over a GRE tunnel:

- RFC 1191, Path MTU Discovery
- RFC 4459, MTU and Fragmentation Issues with In-the-Network Tunneling

RFC 1191 describes a method for dynamically discovering the maximum transmission unit (MTU) of an arbitrary internet path. When a FastIron device receives an IP packet that has its Do not Fragment (DF) bit set, and the packet size is greater than the MTU value of the outbound interface, then the FastIron device returns an ICMP Destination Unreachable message to the source of the packet, with the code indicating "fragmentation needed and DF set". The ICMP Destination Unreachable message includes the MTU of the outbound interface. The source host can use this information to help determine the minimum MTU of a path to a destination.

RFC 4459 describes solutions for issues with large packets over a tunnel. The following methods, from RFC 4459, are supported in Brocade IronWare software:

- If a source attempts to send packets that are larger than the lowest MTU value along the path, PMTUD can signal to the source to send smaller packets. This method is described in Section 3.2 of RFC 4459.
- Inner packets can be fragmented before encapsulation, in such a manner that the encapsulated packet fits in the tunnel path MTU, which is discovered using PMTUD. This method is described in Section 3.4 of RFC 4459.

By default, PMTUD is enabled.

Configuration considerations for PMTUD support

Consider the following when configuring PMTUD support.

- When the new PMTUD value is smaller than all of the eight MTU values configured in the system, the PMTUD feature is disabled for the tunnel, and the value is not added to the system. For example, the new PMTUD value is 620 which is smaller in value than all of the eight, different MTU path values configured in the system. The following warning message is displayed on the CLI:

  ```
  Warning - All MTU profiles used, disabling PMTU for tunnel <tunnel_id>; new
  PMTU was <new pmtu discovered>
  ```

Support for IPv4 multicast routing over GRE tunnels

PIM-DM and PIM-SM Layer 3 multicast protocols and multicast data traffic are supported over GRE tunnels. When a multicast protocol is enabled on both ends of a GRE tunnel, multicast packets can be sent from one tunnel endpoint to another. To accomplish this, the packets are encapsulated using the GRE unicast tunneling mechanism and forwarded like any other IPv4 unicast packet to the destination endpoint of the tunnel. The router that terminates the tunnel (i.e., the router where the tunnel endpoint is an ingress interface) de-encapsulates the GRE tunneled packet to retrieve the native multicast data packets. After de-encapsulation, data packets are forwarded in the direction of its receivers, and control packets may be consumed. This creates a PIM-enabled virtual or logical link between the two GRE tunnel endpoints.

Strict RPF check for multicast protocols

IronWare software enforces strict Reverse Path Forwarding (RPF) check rules on an (s,g) entry on a GRE tunnel interface. The (s,g) entry uses the GRE tunnel as an RPF interface. During unicast routing transit, GRE tunnel packets may arrive at different physical interfaces. The strict RPF check limits GRE PIM tunnel interfaces to accept the (s,g) GRE tunnel traffic.

NOTE
For the Brocade ICX 6650 devicesloopback ports are required for de-encapsulating the GRE tunneled packet. On these hardware devices, when the GRE-encapsulated multicast packet is received, the unicast GRE mechanism takes care of de-encapsulating the packet. The packet then egresses and re-ingresses the tunnel interface loopback port as the native multicast packet. The hardware RPF check is done, not on the tunnel interface directly, but on the loopback port - the hardware compares this port number with the port number configured in the Multicast table (s,g) entry. If they match, the packet is routed. Otherwise it is sent to the CPU for error processing. In

unicast, it is permissible for multiple tunnel interfaces to use a single loopback port. However, in multicast, this will not allow the hardware to determine the tunnel interface that the packet was received on in order to do an RPF check. Therefore, when IPv4 Multicast Routing is enabled on a GRE tunnel, the tunnel interface must have a dedicated loopback port.

GRE support with other features

This section describes how GRE tunnels may affect other features on Brocade ICX 6650 devices.

Support for ECMP for routes through a GRE tunnel

Equal-Cost Multi-Path (ECMP) load sharing allows for load distribution of traffic among available routes. When GRE is enabled, a mix of GRE tunnels and normal IP routes is supported. If multiple routes are using GRE tunnels to a destination, packets are automatically load-balanced between tunnels, or between tunnels and normal IP routes.

ACL, QoS, and PBR support for traffic through a GRE tunnel

PBR and ACL filtering for packets terminating on a GRE tunnel is not supported. However, PBR can be used to map IP traffic into a GRE tunnel, but it cannot be used to route GRE traffic. QoS support for GRE encapsulated packets is limited to copying DSCP values from the inner header onto the outer headerTraffic coming from a tunnel can be filtered by an ACL both before and after the tunnel is terminated and also redirected by PBR after tunnel is terminated. An ACL classifies and sets QoS for GRE traffic. If the ACL is applied to the tunnel ingress port, then the delivery header (outer header) would be classified or filtered before the tunnel is terminated.

NOTE
Restrictions for using ACLs in conjunction with GRE are noted in the section "Configuration considerations for GRE IP tunnels" on page 98. PBR can be configured on tunnel loopback ports for tunnel interfaces with no restrictions. .

Syslog messages related to GRE IP tunnels

Syslog messages provide management applications with information related to GRE IP tunnels. The following Syslog message is supported.

```
Tunnel: TUN-RECURSIVE-DOWN tnnl 1, Tnl disabled due to recursive routing
```

Configuration considerations for GRE IP tunnels

Before configuring GRE tunnels and tunnel options, consider the configuration notes in this section.

- When GRE is enabled on a Layer 3 switch, the following features are not supported on Virtual Ethernet (VE) ports and VE member ports (ports that have IP addresses):
  - ACL logging
  - ACL statistics (also called ACL counting)
  - MAC address filters
  - IPv6 filters

NOTE
The above features are supported on VLANs that do not have VE ports.

- Whenever multiple IP addresses are configured on a tunnel source, the primary address of the tunnel is always used for forming the tunnel connections. Therefore, carefully check the configurations when configuring the tunnel destination.
- When a GRE tunnel is configured, you cannot configure the same routing protocol on the tunnel through which you learn the route to the tunnel destination. For example, if the FastIron learns the tunnel destination route through the OSPF protocol, you cannot configure the OSPF protocol on the same tunnel and vice-versa. When a tunnel has OSPF configured, the FastIron cannot learn the tunnel destination route through OSPF. This could cause the system to become unstable.
- The tunnel destination cannot be resolved to the tunnel itself or any other local tunnel. This is called recursive routing. This scenario would cause the tunnel interface to flap and the Syslog message TUN-RECURSIVE-DOWN to be logged. To resolve this issue, create a static route for the tunnel destination.

GRE MTU configuration considerations

The default Maximum Transmission Unit (MTU) value for packets in a GRE tunnel is 1476 bytes, or 10194 bytes for jumbo packets. The MTU of the GRE tunnel is compared with the outgoing packet before the packet is encapsulated. After encapsulation, the packet size increases by 24 bytes. Therefore, when changing the GRE tunnel MTU, set the MTU to at least 24 bytes less than the IP MTU of the outgoing interface. If the MTU is not set to at least 24 bytes less than the IP MTU, the size of the encapsulated packet will exceed the IP MTU of the outgoing interface. This will cause the packet to either be sent to the CPU for fragmentation, or the packet will be dropped if the DF (Do-Not-Fragment) bit is set in the original IP packet, and an ICMP message is sent.

Configuration tasks for GRE tunnels

Brocade recommends that you perform the configuration tasks in the order listed in Table 13.

TABLE 13 Configuration tasks for GRE tunnels

Configuration tasks		Default behavior	For more information
Required tasks			
1	Create a tunnel interface	Not assigned	"Creating a tunnel interface" on page 101
2	Configure the source address or source interface for the tunnel interface	Not assigned	"Configuring the source address or source interface for a tunnel interface" on page 101
3	Configure the destination address of the tunnel interface	Not assigned	"Configuring the destination address for a tunnel interface" on page 102
4	Enable GRE encapsulation on the tunnel interface	Disabled	"Enabling GRE encapsulation on a tunnel interface" on page 102
5	Configure an IP address for the tunnel interface	Not assigned	"Configuring an IP address for a tunnel interface" on page 103
6	If a route to the tunnel destination (configured in Step 3) does not already exist, create a static route to the tunnel destination.	Not assigned	"Configuring a static route to a tunnel destination" on page 103
Optional tasks			
1	Change the maximum transmission unit (MTU) value for the tunnel interface	1476 bytes or 10194 bytes (jumbo mode)	"Changing the MTU value for a tunnel interface" on page 104
2	Change the number of GRE tunnels supported on the device	Support for 32 GRE tunnels	"Changing the maximum number of tunnels supported" on page 104
3	Enable and configure GRE link keepalive on the tunnel interface	Disabled	"Configuring GRE link keepalive" on page 104
4	Change the Path MTU Discovery (PMTUD) configuration on the GRE tunnel interface	Enabled	"Configuring Path MTU Discovery" on page 105
5	Enable support for IPv4 multicast routing	Disabled	"Enabling IPv4 multicast routing over a GRE tunnel" on page 106

The following features are also supported on GRE tunnel interfaces:

- Naming the tunnel interface (CLI command **port-name**) – for configuration details, refer to the section "Assigning a port name" in the *Brocade ICX 6650 Administration Guide*.
- Changing the Maximum Transmission Unit (MTU) (CLI command **ip mtu**) – for configuration details, refer to "Changing the MTU on an individual port" on page 30.
- Increasing the cost of routes learned on the port (CLI command **ip metric**) – for configuration details, refer to "Changing the cost of routes learned on a port" on page 144.

After performing the configuration steps listed in Table 13, you can view the GRE configuration and observe the routes that use GRE tunnels. For details, refer to "Displaying GRE tunneling information" on page 108.

Creating a tunnel interface

To create a tunnel interface, enter the following command at the Global CONFIG level of the CLI.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)#
```

Syntax: [**no**] **interface tunnel** *tunnel-number*

The *tunnel-number* is a numerical value that identifies the tunnel being configured.

NOTE
You can also use the **port-name** command to name the tunnel. To do so, follow the configuration instructions in the “Assigning a port name” section in the *Brocade ICX 6650 Administration Guide*.

Configuring the source address or source interface for a tunnel interface

To configure the source for a tunnel interface, specify either a source address or a source interface.

NOTE
If the destination address for a tunnel interface is not resolved, Brocade recommends that you either configure *source interface* (instead of *source address*) as the source for a tunnel interface, or enable GRE link keepalive on the tunnel interface.

The tunnel ***source address*** should be one of the router IP addresses configured on a physical, loopback, or VE interface, through which the other end of the tunnel is reachable.

To configure the source address for a specific tunnel interface, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel source 192.168.10.8
```

The ***source interface*** should be the port number of the interface configured on a physical, loopback, or VE interface. The source interface should have at least one IP address configured on it. Otherwise, the interface will not be added to the tunnel configuration and an error message similar to the following will be displayed:

```
    ERROR - Tunnel source interface 1/1/3 has no configured IP address.
```

To configure the source interface for a specific tunnel interface, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel source ethernet 1/1/3
```

Syntax: [**no**] **tunnel source** *ip-address* | **ethernet** *portnum* | **ve** *number*

The *ip-address* variable is the source IP address being configured for the specified tunnel.

The ethernet *portnum* variable is the stack unit, source slot and port number of the physical interface being configured for the specified tunnel, for example 1/1/3.

The ve *number* variable is the VE interface number being configured for the specified tunnel.

Deleting an IP address from an interface configured as a tunnel source

To delete an IP address from an interface that is configured as a tunnel source, first remove the tunnel source from the tunnel interface then delete the IP address, as shown in the following example.

```
Brocade(config-if-e1000-1/1/3)# interface tunnel 8
Brocade(config-tnif-8)# no tunnel source 192.168.83.15
Brocade(config-tnif-8)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# no ip address 192.168.83.15/24
```

If you attempt to delete an IP address without first removing the tunnel source, the console will display an error message, as shown in the following example.

```
Brocade# config terminal
Brocade(config)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# no ip address 192.168.83.15/24
Error - Please remove tunnel source from tnnl 8 before removing IP address
```

NOTE
The previous error message will also display on the CLI when an interface is part of a VLAN. A VLAN cannot be deleted until the tunnel source is first removed.

Configuring the destination address for a tunnel interface

The destination address should be the address of the IP interface of the device on the other end of the tunnel.

To configure the destination address for a specific tunnel interface, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel destination 192.168.5.2
```

Syntax: [no] **tunnel destination** *ip-address*

The *ip-address* variable is the destination IP address being configured for the specified tunnel.

NOTE
Ensure a route to the tunnel destination exists on the tunnel source device. Create a static route if necessary. For configuration details, refer to "Configuring a static route to a tunnel destination" on page 103.

Enabling GRE encapsulation on a tunnel interface

To enable GRE encapsulation on a tunnel interface, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel mode gre ip
```

Syntax: [no] **tunnel mode gre ip**

- **gre** specifies that the tunnel will use GRE encapsulation (IP protocol 47).
- **ip** specifies that the tunneling protocol is IPv4.

NOTE
Before configuring a new GRE tunnel, the system should have at least one slot available for adding the default tunnel MTU value to the system tables. Depending on the configuration, the default tunnel MTU range is ((1500 or 10218) - 24) . To check for slot availability, or to see if the MTU value is already configured in the IP table, use the **show ip mtu** command. For more information on the **show ip mtu** command, refer to "Displaying multicast protocols and GRE tunneling information" on page 110.

Applying an ACL or PBR to a tunnel interface

To apply an ACL or PBR policy to a tunnel interface on a Brocade ICX 6650 device , enter commands such as the following:

Applying a PBR policy to a tunnel interface

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel mode gre ip
Brocade(config-tnif-1)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# ip policy route-map test-route
```

Applying an ACL policy to a tunnel interface

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel mode gre ip
Brocade(config-tnif-1)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# ip access-group 10 in
```

Configuring an IP address for a tunnel interface

An IP address sets a tunnel interface as an IP port and allows the configuration of Layer 3 protocols, such as OSPF, BGP, and Multicast (PIM-DM and PIM-SM) on the port. Note that the subnet cannot overlap other subnets configured on other routing interfaces, and both ends of the tunnel should be in the same subnet, as illustrated in the GRE tunnel configuration example in Figure 16 on page 107.

To configure an IP address for a specified tunnel interface, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# ip address 10.10.3.1/24
```

Syntax: [**no**] **ip address** *ip-address*

The *ip-address* is the IP address being configured for the specified tunnel interface.

Configuring a static route to a tunnel destination

If a route to the tunnel destination does not already exist on the tunnel source, create a static route.

```
Brocade(config)# ip route 192.168.5.0/24 192.168.8.1
```

Syntax: [**no**] **ip route** *destination-ip/network next-hop-ip*

- The *destination-ip/netowork* variable is the destination IP address of the tunnel interface.

Changing the MTU value for a tunnel interface

For important configuration considerations regarding this feature, refer to “GRE MTU configuration considerations” on page 99.

You can set an MTU value for packets entering the tunnel. Packets that exceed either the default MTU value of 1476/10194 bytes (for jumbo case) or the value that you set using this command, are fragmented and encapsulated with IP/GRE headers for transit through the tunnel (if they do not have the DF bit set in the IP header). All fragments will carry the same DF bit as the incoming packet. Jumbo packets are supported, although they may be fragmented based on the configured MTU value.

The following command allows you to change the MTU value for packets transiting “tunnel 1”:

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# ip mtu 1200
```

Syntax: **ip mtu** *packet-size*

The *packet-size* variable specifies the maximum size in bytes for the packets transiting the tunnel. Enter a value from 576 through 1476. The default value is 1476.

NOTE
To prevent packet loss after the 24 byte GRE header is added, make sure that any physical interface that is carrying GRE tunnel traffic has an IP MTU setting at least 24 bytes greater than the tunnel MTU setting. This configuration is only allowed on the system if the tunnel mode is set to GRE.

Changing the maximum number of tunnels supported

By default, Brocade ICX 6650 IPv6 devices support up to 32 GRE tunnels. You can configure the device to support 16–64 GRE tunnels. To change the maximum number of tunnels supported, enter commands such as the following.

```
Brocade(config)# system-max gre-tunnels 16
Reload required.  Please write memory and then reload or power cycle.
Brocade(config)# write memory
Brocade(config)# exit
Brocade# reload
```

NOTE
You must save the configuration (write memory) and reload the software to place the change into effect.

Syntax: **system-max gre-tunnels** *number*

The *number* variable specifies the number of GRE tunnels that can be supported on the device. The permissible range is 16–64. The **system-max gre-tunnels** command determines the interface range that is supported for an interface tunnel. For example, if the system-max value is reduced, it is possible that the configured interfaces may be rejected after a system reload.

Configuring GRE link keepalive

When GRE tunnels are used in combination with static routing or policy-based routing, and a dynamic routing protocol such as RIP, BGP, or OSPF is not deployed over the GRE tunnel, a configured tunnel does not have the ability to bring down the line protocol of either tunnel endpoint, if the far end becomes unreachable. Traffic sent on the tunnel cannot follow alternate paths because the tunnel is always UP. To avoid this scenario, enable GRE link keepalive, which will maintain or place the tunnel in an UP or DOWN state based upon the periodic sending of keepalive packets and the monitoring of responses to the packets. If the packets fail to reach the tunnel far end more frequently than the configured number of retries, the tunnel is placed in the DOWN state.

To enable GRE link keepalive, configure it on one end of the tunnel and ensure the other end of the tunnel has GRE enabled.

To configure GRE link keepalive, enter commands such as the following.

```
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# keepalive 12 4
```

These commands configure the device to wait for 4 consecutive lost keepalive packets before bringing the tunnel down. There will be a 12 second interval between each packet. Note that when the tunnel comes up, it would immediately (within one second) send the first keepalive packet.

Syntax: [**no**] **keepalive** *seconds retries*

Use the **no** form of the command to disable the keepalive option.

The *seconds* variable specifies the number of seconds between each initiation of a keepalive message. The range for this interval is 2–32767 seconds. The default value is 10 seconds.

The *retries* variable specifies the number of times that a packet is sent before the system places the tunnel in the DOWN state. Possible values are from 1 through 255. The default number of retries is 3.

Use the **show interface tunnel** and **show ip tunnel traffic** commands to view the GRE link keepalive configuration,. For details, refer to “Displaying GRE tunneling information” on page 108.

Configuring Path MTU Discovery

Path MTU Discovery (PMTUD) support is described in the section “Path MTU Discovery (PMTUD) support” on page 96. PMTUD is enabled by default on tunnel interfaces. This section describes how to disable and re-enable PMTUD on a tunnel interface, change the PMTUD age timer, manually clear the tunnel PMTUD, and view the PMTUD configuration.

Disabling and re-enabling PMTUD

PMTUD is enabled by default. To disable it, enter the following command:

```
Brocade(config-tnif-1)# tunnel path-mtu-discovery disable
```

To re-enable PMTUD after it has been disabled, enter the following command:

```
Brocade(config-tnif-1)# no tunnel path-mtu-discovery disable
```

Syntax: [**no**] **tunnel path-mtu-discovery disable**

Changing the age timer for PMTUD

By default, when PMTUD is enabled on a tunnel interface, the path MTU is reset to its original value every 10 minutes.   If desired, you can change the reset time (default age timer) to a value of up to 30 minutes. To do so, enter a command such as the following on the GRE tunnel interface.

```
Brocade(config-tnif-1)# tunnel path-mtu-discovery age-timer 20
```

This command configures the device to wait for 20 minutes before resetting the path MTU to its original value.

Syntax: [no] tunnel path-mtu-discovery age-timer *minutes* | **infinite**

For *minutes*, enter a value from 10 to 30.

Enter **infinite** to disable the timer.

Clearing the PMTUD dynamic value

To reset a dynamically-configured MTU on a tunnel Interface back to the configured value, enter the following command.

```
Brocade(config)# clear ip tunnel pmtud 1
```

Syntax: clear ip tunnel pmtud *tunnel-ID*

The *tunnel-ID* variable is a valid tunnel number or name.

Viewing PMTUD configuration details

Use the **show interface tunnel** command to view the PMTUD configuration and to determine whether PMTUD has reduced the size of the MTU. For details about the **show interface tunnel** command, refer to "Displaying GRE tunneling information" on page 108.

Enabling IPv4 multicast routing over a GRE tunnel

This section describes how to enable IPv4 multicast protocols, PIM Sparse (PIM-SM) and PIM Dense (PIM-DM), on a GRE tunnel. Perform the procedures in this section after completing the required tasks in Table 13 on page 100.

For an overview of multicast routing support over a GRE tunnel, refer to "Support for IPv4 multicast routing over GRE tunnels" on page 97. To view information about multicast protocols and GRE tunnel-specific information, refer to "Displaying multicast protocols and GRE tunneling information" on page 110.

Enabling PIM-SM on a GRE tunnel

To enable PIM-SM on a GRE tunnel interface, enter commands such as the following:

```
Brocade(config)# interface tunnel 10
Brocade(config-tnif-10)# ip pim-sparse
```

Syntax: [no] ip pim-sparse

Use the **no** form of the command to disable PIM-SM on the tunnel interface.

Enabling PIM-DM on a GRE tunnel interface

To enable PIM-DM on a GRE tunnel interface, enter commands such as the following:

```
Brocade(config)# interface tunnel 10
Brocade(config-tnif-10)# ip pim
```

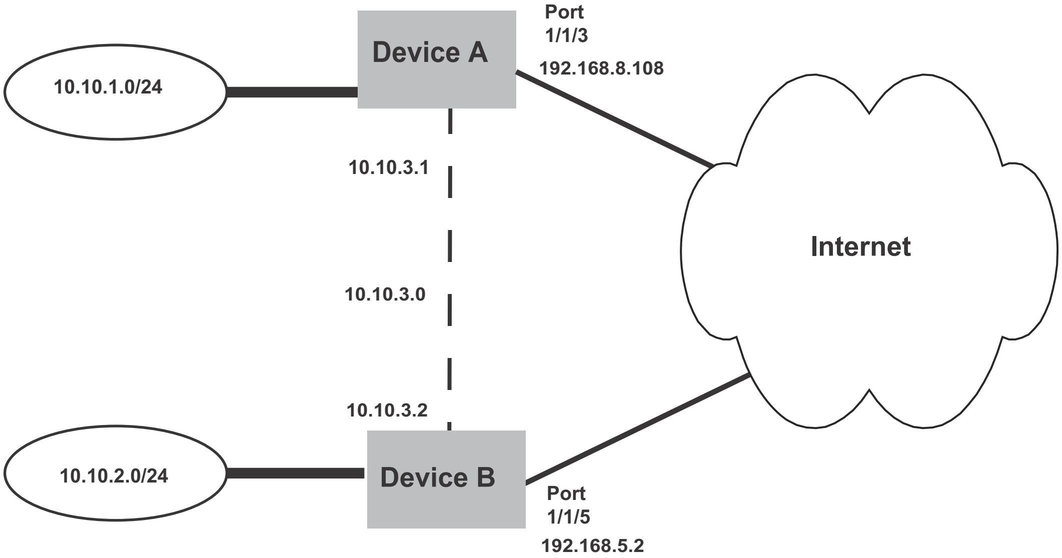
Syntax: [no] ip pim

Use the **no** form of the command to disable PIM-DM on the tunnel interface.

Point-to-point GRE tunnel configuration example

In the configuration example shown in Figure 16, a GRE Tunnel is configured between device A and device B. Traffic between networks 10.10.1.0/24 and 10.10.2.0/24 is encapsulated in a GRE packet sent through the tunnel on the 10.10.3.0 network, and unpacked and sent to the destination network. A static route is configured at each Layer 3 switch to go through the tunnel interface to the target network.

FIGURE 16 **Point-to-point GRE tunnel configuration example**



The following shows the configuration commands for the example shown in Figure 16.

Configuring point-to-point GRE tunnel for device A

```
Brocade (config)# interface ethernet 1/1/3
Brocade (config-if-e10000-1/1/3)# ip address 192.168.8.108/24
Brocade (config)# exit
Brocade (config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel source 192.168.8.108
Brocade(config-tnif-1)# tunnel destination 192.168.5.2
Brocade(config-tnif-1)# tunnel mode gre ip
Brocade(config-tnif-1)# ip address 10.10.3.1/24
Brocade(config-tnif-1)# exit
Brocade (config)# ip route 192.168.5.0/24 192.168.8.1
```

Configuring point-to-point GRE tunnel for device B

```
Brocade(config)# interface ethernet 1/1/5
Brocade(config-if-e10000-1/1/5)# ip address 192.168.5.2/24
Brocade(config)# exit
Brocade(config)# interface tunnel 1
Brocade(config-tnif-1)# tunnel source 192.168.5.2
Brocade(config-tnif-1)# tunnel destination 192.168.8.108
Brocade(config-tnif-1)# tunnel mode gre ip
Brocade(config-tnif-1)# ip address 10.10.3.2/24
Brocade(config-tnif-1)# exit
Brocade(config)# ip route 192.168.8.0/24 192.168.5.1
```

Displaying GRE tunneling information

This section describes the **show** commands that display the GRE tunnels configuration, the link status of the GRE tunnels, and the routes that use GRE tunnels. To display information about multicast protocols and GRE tunnels, refer to "Displaying multicast protocols and GRE tunneling information" on page 110.

To display GRE tunneling Information, use the following commands:

- **show ip interface**
- **show ip route**
- **show ip interface tunnel**
- **show ip tunnel traffic**
- **show interface tunnel**
- **show statistics tunnel**

The following shows an example output of the **show ip interface** command, which includes information about GRE tunnels.

```
Brocade# show ip interface
   Interface       IP-Address     OK?  Method    Status   Protocol
   Tunnel 1        10.10.3.1      YES  NVRAM     up       up
```

For field definitions, refer to Table 17 on page 117.

Syntax: **show ip interface**

The **show ip route** command displays routes that are pointing to a GRE tunnel as shown in the following example.

```
Brocade# show ip route
Total number of IP routes: 3, avail: 79996 (out of max 80000)
B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
        Destination     NetMask         Gateway         Port       Cost   Type
1       192.168.1.0     255.255.255.0   0.0.0.0         7          1      D
2       192.168.2.0     255.255.255.0   0.0.0.3         7          1      S
3       192.168.3.0     255.255.255.0   0.0.0.0         tn3        1      D
```

For field definitions, refer to Table 21 on page 124.

Syntax: **show ip route**

The **show ip interface tunnel** command displays the link status and IP address configuration for an IP tunnel interface as shown in the following example.

```
Brocade# show ip interface tunnel 1
Interface Tunnel 1
  port state: UP
  ip address: 192.168.21.2 subnet mask: 255.255.255.0
  encapsulation: GRE, mtu: 1476, metric: 1
  directed-broadcast-forwarding: disabled
  proxy-arp: disabled
  ip arp-age:  10 minutes
  No Helper Addresses are configured
  No inbound ip access-list is set
  No outgoing ip access-list is set
```

Syntax: **show ip interface tunnel** [*tunnel-ID*]

The *tunnel-ID* variable is a valid tunnel number or name.

The **show interface tunnel** command displays the GRE tunnel configuration.

```
Brocade# show int tunnel 3
Tunnel3 is up, line protocol is up
  Hardware is Tunnel
  Tunnel source  192.168.1.1
  Tunnel destination is 192.168.2.3
    Port name is customer1001
  Internet address is 10.34.3.2/24, MTU 1476 bytes, encapsulation GRE
  Keepalive is Enabled : Interval 10, No.of Retries 3
  Total Keepalive Pkts Tx: 2, Rx: 2
  Path MTU Discovery: Enabled, MTU is 1476 bytes
```

Syntax: **show interface tunnel** [*tunnel-ID*]

This display shows the following information.

TABLE 14 CLI display of **show interface tunnel** command

Field	Definition
Hardware is Tunnel	The interface is a tunnel interface.
Tunnel source	The source address for the tunnel.
Tunnel destination	The destination address for the tunnel.
Port name	The port name (if applicable).
Internet address	The internet address.
MTU	The maximum transmission unit.
encapsulation GRE	GRE encapsulation is enabled on the port.
Keepalive	Indicates whether or not GRE link keepalive is enabled.
Interval	If GRE link keepalive is enabled, this field displays the number of seconds between each initiation of a GRE link keepalive message.
No. of Retries	If GRE link keepalive is enabled, this field displays the number of times that a GRE link keepalive packet is sent before the tunnel is placed in the DOWN state.
Total Keepalive Pkts	If GRE link keepalive is enabled, this field shows the total number of GRE link keepalive packets transmitted (TX) and received (RX) on the tunnel since it was last cleared by the administrator.
Path MTU Discovery	Indicates whether or not PMTUD is enabled. If PMTUD is enabled, the MTU value is also displayed.

The **show ip tunnel traffic** command displays the link status of the tunnel and the number of keepalive packets received and sent on the tunnel.

```
Brocade# show ip tunnel traffic
IP GRE Tunnels
      Tunnel Status  Packet Received  Packet Sent  KA recv  KA sent
  1   up/up          362              0            362      362
  3   up/up          0                0            0        0
  10  down/down      0                0            0        0
```

Syntax: **show ip tunnel traffic**

The **show statistics tunnel** [*tunnel-ID*] command displays GRE tunnel statistics for a specific tunnel ID number. The following shows an example output for tunnel ID 1.

Syntax: **show statistics tunnel** [*tunnel-ID*]

The *tunnel-ID* variable specifies the tunnel ID number.

This display shows the following information.

TABLE 15 CLI display of **show ip tunnel traffic** command

Field	Description
Tunnel Status	Indicates whether the tunnel is up or down. Possible values are: • Up/Up – The tunnel and line protocol are up. • Up/Down – The tunnel is up and the line protocol is down. • Down/Up – The tunnel is down and the line protocol is up. • Down/Down – The tunnel and line protocol are down.
Packet Received	The number of packets received on the tunnel since it was last cleared by the administrator.
Packet Sent	The number of packets sent on the tunnel since it was last cleared by the administrator.
KA recv	The number of keepalive packets received on the tunnel since it was last cleared by the administrator.
KA sent	The number of keepalive packets sent on the tunnel since it was last cleared by the administrator.

Displaying multicast protocols and GRE tunneling information

The following **show** commands display information about multicast protocols and GRE tunnels:

- **show ip pim interface**
- **show ip pim nbr**
- **show ip pim mcache**
- **show ip pim flow**
- **show statistics**
- **show ip mtu**

NOTE
All other **show** commands that are supported currently for Ethernet, VE, and IP loopback interfaces, are also supported for tunnel interfaces. To display information for a tunnel interface, specify the tunnel in the format **tn** *num*. For example, **show interface tn 1**. In some cases, the Ethernet port that the tunnel is using will be displayed in the format **tn** *num*:**e** *port*.

The following shows an example output of the **show ip pim interface** command. The lines in bold highlight the GRE tunnel-specific information.

```
Brocade# show ip pim interface
Interface e1/1/1
PIM Dense: V2
TTL Threshold: 1, Enabled, DR: itself
Local Address: 10.10.10.10

Interface tn1
PIM Dense: V2
TTL Threshold: 1, Enabled, DR: 10.1.1.20 on tn1:e2
Local Address: 10.1.1.10
Neighbor:
  10.1.1.20
```

Syntax: **show ip pim interface**

The following shows an example output of the **show ip pim nbr** command. The line in bold shows the GRE tunnel-specific information.

```
Brocade# show ip pim nbr
Total number of neighbors: 1 on 1 ports
Port   Phy_p      Neighbor        Holdtime Age   UpTime
tn1    tn1:e2     10.1.1.20        180      60    1740
```

Syntax: **show ip pim nbr**

The following shows an example output of the **show ip pim mcache** command. The line in bold shows the GRE tunnel-specific information.

```
Brocade# show ip pim mcache 192.168.1.1
1    (10.10.10.1 192.168.1.1) in e1/1/1 (e1/1/1), cnt=629
     Source is directly connected
     L3 (HW) 1: tn1:e2(VL1)
     fast=1 slow=0 pru=1 graft
     age=120s up-time=8m HW=1 L2-vidx=8191 has mll
```

Syntax: **show ip pim mcache** *ip-address*

The following shows an example output of the **show ip pim flow** command. The text in bold highlights the GRE tunnel-specific information.

```
Brocade# show ip pim flow 192.168.1.1

Multicast flow (10.10.10.1 192.168.1.1):
 Vidx for source vlan forwarding: 8191 (Blackhole, no L2 clients)
 Hardware MC Entry hit on devices: 0 1 2 3
 MC Entry[0x0c008040]: 00014001 000022ee 0ffc0001 00000000
 --- MLL contents read from Device 0 ---
 MLL Data[0x018c0010]: 0021ff8d 00000083 00000000 00000000
 First  : Last:1, outlif:60043ff1 00000000, TNL:1(e2)

1 flow printed
```

Syntax: **show ip pim flow**

The following shows an example output of the **show statistics** command. The following statistics demonstrate an example where the encapsulated multicast traffic ingresses a tunnel endpoint on port e 1/1/2, egresses and re-ingresses as native multicast traffic on the loopback port e 1/1/4, and is then forwarded to the outbound interface e 1/1/1.

```
Brocade# show statistics

Port         In Packets          Out Packets       In Errors      Out Errors
1                     0                 1670               0               0
2                  1668                    7               0               0
3                     0                    0               0               0
4                  1668                 1668               0               0
```

Syntax: **show statistics**

The **show ip mtu** command can be used to see if there is space available for the ip_default_mtu_24 value in the system, or if the MTU value is already configured in the IP table. The following shows an example output of the **show ip mtu** command.

```
  Brocade(config-tnif-10)#show ip mtu
  idx   size  usage  ref-count
    0  10218      1    default
    1    800      0          1
    2    900      0          1
    3    750      0          1
    4  10194      1          1
    5  10198      0          1
```

Syntax: **show ip mtu**

Clearing GRE statistics

Use the **clear ip tunnel** command to clear statistics related to GRE tunnels.

To clear GRE tunnel statistics, enter a command such as the following.

```
Brocade(config)# clear ip tunnel stat 3
```

To reset a dynamically-configured MTU on a tunnel Interface back to the configured value, enter a command such as the following.

```
Brocade(config)#clear ip tunnel pmtud 3
```

Syntax: **clear ip tunnel** [**pmtud** *tunnel-ID* | **stat** *tunnel-ID*]

Use the **pmtud** option to reset a dynamically-configured MTU on a tunnel Interface back to the configured value.

Use the **stat** option to clear tunnel statistics.

The *tunnel-ID* variable is a valid tunnel number or name.

Use the **clear statistics tunnel** [*tunnel-ID*] command to clear GRE tunnel statistics for a specific tunnel ID number. To clear GRE tunnel statistics for tunnel ID 3, enter a command such as the following.

```
Brocade(config)# clear statistics tunnel 3
```

Syntax: **clear statistics tunnel** [*tunnel-ID*]

The *tunnel-ID* variable specifies the tunnel ID number.

Displaying IP configuration information and statistics

The following sections describe IP display options for Layer 3 Switches and Layer 2 Switches:

- To display IP information on a Layer 3 Switch, refer to "Displaying IP information – Layer 3 Switches" on page 113.
- To display IP information on a Layer 2 Switch, refer to "Displaying IP information – Layer 2 Switches" on page 128.

Changing the network mask display to prefix format

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the displays to prefix format (example: /18) on a Layer 3 Switch or Layer 2 Switch using the following CLI method.

To enable CIDR format for displaying network masks, entering the following command at the global CONFIG level of the CLI.

```
Brocade(config)# ip show-subnet-length
```

Syntax: [no] **ip show-subnet-length**

Displaying IP information – Layer 3 Switches

You can display the following IP configuration information statistics on Layer 3 Switches:

- Global IP parameter settings and IP access policies – refer to "Displaying global IP configuration information" on page 113.
- CPU utilization statistics – refer to "Displaying CPU utilization statistics" on page 115.
- IP interfaces – refer to "Displaying IP interface information" on page 117.
- ARP entries – refer to "Displaying ARP entries" on page 118.
- Static ARP entries – refer to "Displaying ARP entries" on page 118.
- IP forwarding cache – refer to "Displaying the forwarding cache" on page 121.
- IP route table – refer to "Displaying the IP route table" on page 122.
- IP traffic statistics – refer to "Displaying IP traffic statistics" on page 125.

The following sections describe how to display this information.

In addition to the information described below, you can display the following IP information. This information is described in other parts of this guide:

- RIP
- OSPF
- BGP4
- PIM
- VRRP or VRRP-E

Displaying global IP configuration information

To display IP configuration information, enter the following command at any CLI level.

```
Brocade# show ip

Global Settings
  ttl: 64, arp-age: 10, bootp-relay-max-hops: 4
  router-id :10.95.11.128
  enabled : UDP-Broadcast-Forwarding  IRDP  Proxy-ARP  RARP  OSPF
  disabled: BGP4 Load-Sharing  RIP DVMRP FSRP  VRRP
Static Routes
  Index   IP Address        Subnet Mask       Next Hop Router   Metric Distance
  1       0.0.0.0           0.0.0.0           10.157.23.2      1      1
Policies
  Index   Action   Source           Destination      Protocol   Port  Operator
  1       deny     10.157.22.34    10.157.22.26    tcp        http  =
  64      permit   any              any
```

Syntax: show ip

NOTE
This command has additional options, which are explained in other sections in this guide, including the sections following this one.

This display shows the following information.

TABLE 16 CLI display of global IP configuration information – Layer 3 Switch

Field	Description
Global settings	
ttl	The Time-To-Live (TTL) for IP packets. The TTL specifies the maximum number of router hops a packet can travel before reaching the Brocade router. If the packet TTL value is higher than the value specified in this field, the Brocade router drops the packet. To change the maximum TTL, refer to "Changing the TTL threshold" on page 41.
arp-age	The ARP aging period. This parameter specifies how many minutes an inactive ARP entry remains in the ARP cache before the router ages out the entry. To change the ARP aging period, refer to "Changing the ARP aging period" on page 37.
bootp-relay-max-hops	The maximum number of hops away a BootP server can be located from the Brocade router and still be used by the router clients for network booting. To change this value, refer to "Changing the maximum number of hops to a BootP relay server" on page 67.
router-id	The 32-bit number that uniquely identifies the Brocade router. By default, the router ID is the numerically lowest IP interface configured on the router. To change the router ID, refer to "Changing the router ID" on page 31.
enabled	The IP-related protocols that are enabled on the router.
disabled	The IP-related protocols that are disabled on the router.
Static routes	
Index	The row number of this entry in the IP route table.
IP Address	The IP address of the route destination.
Subnet Mask	The network mask for the IP address.
Next Hop Router	The IP address of the router interface to which the Brocade router sends packets for the route.
Metric	The cost of the route. Usually, the metric represents the number of hops to the destination.

TABLE 16 CLI display of global IP configuration information – Layer 3 Switch (Continued)

Field	Description
Distance	The administrative distance of the route. The default administrative distance for static IP routes in Brocade routers is 1. To list the default administrative distances for all types of routes or to change the administrative distance of a static route, refer to "Changing the administrative distance" on page 146.
Policies	
Index	The policy number. This is the number you assigned the policy when you configured it.
Action	The action the router takes if a packet matches the comparison values in the policy. The action can be one of the following: • deny – The router drops packets that match this policy. • permit – The router forwards packets that match this policy.
Source	The source IP address the policy matches.
Destination	The destination IP address the policy matches.
Protocol	The IP protocol the policy matches. The protocol can be one of the following: • ICMP • IGMP • IGRP • OSPF • TCP • UDP
Port	The Layer 4 TCP or UDP port the policy checks for in packets. The port can be displayed by its number or, for port types the router recognizes, by the well-known name. For example, TCP port 80 can be displayed as HTTP. NOTE: This field applies only if the IP protocol is TCP or UDP.
Operator	The comparison operator for TCP or UDP port names or numbers. NOTE: This field applies only if the IP protocol is TCP or UDP.

Displaying CPU utilization statistics

You can display CPU utilization statistics for IP protocols using the **show process cpu** command.

The **show process cpu** command includes CPU utilization statistics for ACL, 802.1x, and L2VLAN. L2VLAN contains any packet transmitted to a VLAN by the CPU, including unknown unicast, multicast, broadcast, and CPU forwarded Layer 2 traffic.

To display CPU utilization statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI.

```
Brocade# show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ACL              0.00      0.00      0.00      0.00              0
ARP              0.01      0.01      0.01      0.01            714
BGP              0.00      0.00      0.00      0.00              0
DOT1X            0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00            161
IP               0.00      0.00      0.00      0.00            229
L2VLAN           0.01      0.00      0.00      0.01            673
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              9
STP              0.00      0.00      0.00      0.00              7
VRRP             0.00      0.00      0.00      0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example.

```
Brocade# show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ACL              0.00      0.00      0.00      0.00              0
ARP              0.01      0.01      0.01      0.01            714
BGP              0.00      0.00      0.00      0.00              0
DOT1X            0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00            161
IP               0.00      0.00      0.00      0.00            229
L2VLAN           0.01      0.00      0.00      0.01            673
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              9
STP              0.00      0.00      0.00      0.00              7
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following.

```
Brocade# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ACL                0      0.00
ARP                1      0.01
BGP                0      0.00
DOT1X              0      0.00
GVRP               0      0.00
ICMP               0      0.00
IP                 0      0.00
L2VLAN             1      0.01
OSPF               0      0.00
RIP                0      0.00
STP                0      0.00
VRRP               0      0.00
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: **show process cpu** [*num*]

The *num* parameter specifies the number of seconds and can be from 1 through 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying IP interface information

To display IP interface information, enter the following command at any CLI level.

```
Brocade# show ip interface

Interface         IP-Address     OK?  Method    Status  Protocol
Ethernet 1/1/1    10.95.6.173    YES  NVRAM     up      up
Ethernet 1/1/2    10.3.3.3       YES  manual    up      up
Loopback 1        10.2.3.4       YES  NVRAM     down    down
```

Syntax: **show ip interface** [**ethernet** *stack-unit/slotnum/portnum*] | [**loopback** *num*] |[**tunnel** *num*] | [**ve** *num*]

This display shows the following information.

TABLE 17 CLI display of interface IP configuration information

Field	Description
Interface	The type and the slot and port number of the interface.
IP-Address	The IP address of the interface. **NOTE:** If an "s" is listed following the address, this is a secondary address. When the address was configured, the interface already had an IP address in the same subnet, so the software required the "secondary" option before the software could add the interface.
OK?	Whether the IP address has been configured on the interface.
Method	Whether the IP address has been saved in NVRAM. If you have set the IP address for the interface in the CLI, but have not saved the configuration, the entry for the interface in the Method field is "manual".
Status	The link status of the interface. If you have disabled the interface with the **disable** command, the entry in the Status field will be "administratively down". Otherwise, the entry in the Status field will be either "up" or "down".
Protocol	Whether the interface can provide two-way communication. If the IP address is configured, and the link status of the interface is up, the entry in the protocol field will be "up". Otherwise the entry in the protocol field will be "down".

To display detailed IP information for a specific interface, enter a command such as the following.

```
Brocade# show ip interface ethernet 1/1/1
Interface Ethernet 1/1/1
  port state: UP
  ip address: 192.168.9.51      subnet mask: 255.255.255.0
  encapsulation: ETHERNET, mtu: 1500, metric: 1
  directed-broadcast-forwarding: disabled
  proxy-arp: disabled
  ip arp-age:  10 minutes
  Ip Flow switching is disabled
  No Helper Addresses are configured.
  No inbound ip access-list is set
  No outgoing ip access-list is set
```

Displaying ARP entries

You can display the ARP cache and the static ARP table. The ARP cache contains entries for devices attached to the Layer 3 Switch. The static ARP table contains the user-configured ARP entries. An entry in the static ARP table enters the ARP cache when the entry interface comes up.

The tables require separate display commands.

Displaying the ARP cache

To display the contents of the ARP cache, enter the following command at any CLI level.

```
Brocade# show arp

Total number of ARP entries: 5, maximum capacity: 6000

No.   IP Address        MAC Address         Type       Age     Port   Status
1     10.95.6.102      0000.00fc.ea21      Dynamic     0       1/1/6  Valid
2     10.95.6.18       0000.00d2.04ed      Dynamic     3       1/1/6  Pend
3     10.95.6.54       0000.00ab.cd2b      Dynamic     0       1/1/6  Pend
4     10.95.6.101      0000.007c.a7fa      Dynamic     0       1/1/6  Valid
5     10.95.6.211      0000.0038.ac9c      Dynamic     0       1/1/6  Valid
```

Syntax: **show arp** [**ethernet** *stack-unit/slotnum/portnum* | **mac-address** *xxxx.xxxx.xxxx* [*mask*] | *ip-addr* [*ip-mask*]] [*num*]

The **mac-address** *xxxx.xxxx.xxxx* parameter lets you restrict the display to entries for a specific MAC address.

The *mask* parameter lets you specify a mask for the **mac-address** *xxxx.xxxx.xxxx* parameter, to display entries for multiple MAC addresses. Specify the MAC address mask as “f”s and “0”s, where “f”s are significant bits.

The *ip-addr* and *ip-mask* parameters let you restrict the display to entries for a specific IP address and network mask. Specify the IP address masks in standard decimal mask format (for example, 255.255.0.0).

NOTE
The *ip-mask* parameter and *mask* parameter perform different operations. The *ip-mask* parameter specifies the network mask for a specific IP address, whereas the *mask* parameter provides a filter for displaying multiple MAC addresses that have specific values in common.

The *num* parameter lets you display the table beginning with a specific entry number.

NOTE
The entry numbers in the ARP cache are not related to the entry numbers for static ARP table entries.

This display shows the following information. The number in the left column of the CLI display is the row number of the entry in the ARP cache. This number is not related to the number you assign to static MAC entries in the static ARP table.

TABLE 18 CLI display of ARP cache

Field	Description
Total number of ARP Entries	The number of entries in the ARP cache.
Maximum capacity	The total number of ARP entries supported on the device.
IP Address	The IP address of the device.
MAC Address	The MAC address of the device.
Type	The ARP entry type, which can be one of the following: • Dynamic – The Layer 3 Switch learned the entry from an incoming packet. • Static – The Layer 3 Switch loaded the entry from the static ARP table when the device for the entry was connected to the Layer 3 Switch. • DHCP – The Layer 3 Switch learned the entry from the DHCP binding address table. NOTE: If the type is DHCP, the port number will not be available until the entry gets resolved through ARP.
Age	The number of minutes before which the ARP entry was refreshed. If this value reaches the ARP aging period, the entry is removed from the table. To display the ARP aging period, refer to "Displaying global IP configuration information" on page 113. To change the ARP aging interval, refer to "Changing the ARP aging period" on page 37. NOTE: Static entries do not age out.
Port	The port on which the entry was learned. NOTE: If the ARP entry type is DHCP, the port number will not be available until the entry gets resolved through ARP.
Status	The status of the entry, which can be one of the following: • Valid – This a valid ARP entry. • Pend – The ARP entry is not yet resolved.

Displaying the static ARP table

To display the static ARP table instead of the ARP cache, enter the following command at any CLI level.

```
Brocade# show ip static-arp

Static ARP table size: 512, configurable from 512 to 1024
  Index   IP Address          MAC Address         Port
  1       10.95.6.111        0000.003b.d210      1/1/1
  3       10.95.6.123        0000.003b.d211      1/1/2
```

This example shows two static entries. Note that because you specify an entry index number when you create the entry, it is possible for the range of index numbers to have gaps, as shown in this example.

NOTE
The entry number you assign to a static ARP entry is not related to the entry numbers in the ARP cache.

Syntax: **show ip static-arp** [**ethernet** *stack-unit/slotnum/portnum* | **mac-address** *xxxx.xxxx.xxxx* [*mask*] |
ip-addr [*ip-mask*]] [*num*]

The **mac-address** *xxxx.xxxx.xxxx* parameter lets you restrict the display to entries for a specific MAC address.

The *mask* parameter lets you specify a mask for the **mac-address** *xxxx.xxxx.xxxx* parameter, to display entries for multiple MAC addresses. Specify the MAC address mask as "f"s and "0"s, where "f"s are significant bits.

The *ip-addr* and *ip-mask* parameters let you restrict the display to entries for a specific IP address and network mask. Specify the IP address masks in standard decimal mask format (for example, 255.255.0.0).

NOTE
The *ip-mask* parameter and *mask* parameter perform different operations. The *ip-mask* parameter specifies the network mask for a specific IP address, whereas the *mask* parameter provides a filter for displaying multiple MAC addresses that have specific values in common.

The *num* parameter lets you display the table beginning with a specific entry number.

TABLE 19 CLI display of static ARP table

Field	Description
Static ARP table size	The maximum number of static entries that can be configured on the device using the current memory allocation. The range of valid memory allocations for static ARP entries is listed after the current allocation. To change the memory allocation for static ARP entries, refer to“Changing the maximum number of entries the static ARP table can hold” on page 40.
Index	The number of this entry in the table. You specify the entry number when you create the entry.
IP Address	The IP address of the device.
MAC Address	The MAC address of the device.
Port	The port attached to the device the entry is for.

Displaying the forwarding cache

To display the IP forwarding cache, enter the following command at any CLI level.

```
Brocade# show ip cache

Total number of cache entries: 3
D:Dynamic  P:Permanent  F:Forward  U:Us  C:Complex Filter
W:Wait ARP  I:ICMP Deny  K:Drop  R:Fragment  S:Snap Encap
      IP Address       Next Hop         MAC              Type  Port  Vlan  Pri
1     192.168.1.11     DIRECT           0000.0000.0000   PU    n/a         0
2     192.168.1.255    DIRECT           0000.0000.0000   PU    n/a         0
3     192.168.255.255  DIRECT           0000.0000.0000   PU    n/a         0
```

Syntax: **show ip cache** [*ip-addr*] | [*num*]

The *ip-addr* parameter displays the cache entry for the specified IP address.

The *num* parameter displays the cache beginning with the row following the number you enter. For example, to begin displaying the cache at row 10, enter the following command.

```
Brocade# show ip cache 9
```

The **show ip cache** command displays the following information.

TABLE 20 CLI display of IP forwarding cache – Layer 3 Switch

Field	Description
IP Address	The IP address of the destination.
Next Hop	The IP address of the next-hop router to the destination. This field contains either an IP address or the value DIRECT. DIRECT means the destination is either directly attached or the destination is an address on this Brocade device. For example, the next hop for loopback addresses and broadcast addresses is shown as DIRECT.
MAC	The MAC address of the destination. **NOTE:** If the entry is type U (indicating that the destination is this Brocade device), the address consists of zeroes.

TABLE 20 CLI display of IP forwarding cache – Layer 3 Switch (Continued)

Field	Description
Type	The type of host entry, which can be one or more of the following: • D – Dynamic • P – Permanent • F – Forward • U – Us • C – Complex Filter • W – Wait ARP • I – ICMP Deny • K – Drop • R – Fragment • S – Snap Encap
Port	The port through which this device reaches the destination. For destinations that are located on this device, the port number is shown as “n/a”.
VLAN	Indicates the VLANs the listed port is in.
Pri	The QoS priority of the port or VLAN.

Displaying the IP route table

To display the IP route table, enter the **show ip route** command at any CLI level.

```
Brocade# show ip route
Total number of IP routes: 514
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
Destination        NetMask           Gateway           Port   Cost   Type
Destination        NetMask           Gateway              Port   Cost   Type
10.1.0.0           255.255.0.0       192.168.1.2          1/1/1    2      R
10.2.0.0           255.255.0.0       192.168.1.2          1/1/2    2      R
10.3.0.0           255.255.0.0       192.168.1.2          1/1/3    2      R
10.4.0.0           255.255.0.0       192.168.1.2          1/1/4    2      R
10.5.0.0           255.255.0.0       192.168.1.2          1/1/5    2      R
10.6.0.0           255.255.0.0       192.168.1.2          1/1/6    2      R
10.7.0.0           255.255.0.0       192.168.1.2          1/1/7    2      R
10.8.0.0           255.255.0.0       192.168.1.2          1/1/8    2      R
10.9.0.0           255.255.0.0       192.168.1.2          1/1/9    2      R
10.10.0.0          255.255.0.0       192.168.1.2          1/1/10   2      S
```

Syntax: **show ip route** [*ip-addr* [*ip-mask*] [**longer**] [**none-bgp**]] | *num* | **bgp** | **direct** | **ospf** | **rip** | **static**

The *ip-addr* parameter displays the route to the specified IP address.

The *ip-mask* parameter lets you specify a network mask or, if you prefer CIDR format, the number of bits in the network mask. If you use CIDR format, enter a forward slash immediately after the IP address, then enter the number of mask bits (for example: 10.157.22.0/24 for 10.157.22.0 255.255.255.0).

The **longer** parameter applies only when you specify an IP address and mask. This option displays only the routes for the specified IP address and mask. Refer to the following example.

The **none-bgp** parameter displays only the routes that did not come from BGP4.

The *num* option display the route table entry whose row number corresponds to the number you specify. For example, if you want to display the tenth row in the table, enter “10”.

The **bgp** option displays the BGP4 routes.

The **direct** option displays only the IP routes that are directly attached to the Layer 3 Switch.

The **ospf** option displays the OSPF routes.

The **rip** option displays the RIP routes.

The **static** option displays only the static IP routes.

The **default** routes are displayed first.

Here is an example of how to use the **direct** option. To display only the IP routes that go to devices directly attached to the Layer 3 Switch, enter the following command.

```
Brocade# show ip route direct
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
      Destination       NetMask           Gateway           Port   Cost   Type
      10.157.22.0      255.255.255.0     0.0.0.0           1/1/4   1      D
```

Notice that the route displayed in this example has “D” in the Type field, indicating the route is to a directly connected device.

Here is an example of how to use the **static** option. To display only the static IP routes, enter the following command.

```
Brocade# show ip route static
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
      Destination       NetMask           Gateway           Port   Cost   Type
      192.168.33.11     255.255.255.0     10.157.22.12     1/1/3    2      S
```

Notice that the route displayed in this example has “S” in the Type field, indicating the route is static.

Here is an example of how to use the **longer** option. To display only the routes for a specified IP address and mask, enter a command such as the following.

```
Brocade# show ip route 10.159.0.0/16 longer
Starting index: 1 B:BGP D:Directly-Connected R:RIP S:Static O:OSPF
Destination   NetMask        Gateway       Port  Cost Type
10.159.38.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.39.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.40.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.41.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.42.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.43.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.44.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.45.0   255.255.255.0  10.95.6.101   1/1/1  1     S
10.159.46.0   255.255.255.0  10.95.6.101   1/1/1  1     S
```

This example shows all the routes for networks beginning with 10.159. The mask value and **longer** parameter specify the range of network addresses to be displayed. In this example, all routes within the range 10.159.0.0 – 10.159.255.255 are listed.

The **summary** option displays a summary of the information in the IP route table. The following is an example of the output from this command.

Example

```
Brocade# show ip route summary

IP Routing Table - 35 entries:
  6 connected, 28 static, 0 RIP, 1 OSPF, 0 BGP
  Number of prefixes:
  /0: 1 /16: 27 /22: 1 /24: 5 /32: 1
```

Syntax: **show ip route summary**

In this example, the IP route table contains 35 entries. Of these entries, 6 are directly connected devices, 28 are static routes, and 1 route was calculated through OSPF. One of the routes has a zero-bit mask (this is the default route), 27 have a 16-bit mask, 5 have a 24-bit mask, 1 has 22-bit mask, and 1 has a 32-bit mask.

The following table lists the information displayed by the **show ip route** command.

TABLE 21 **CLI display of IP route table**

Field	Description
Destination	The destination network of the route.
NetMask	The network mask of the destination address.
Gateway	The next-hop router. An asterisk (*) next to the next-hop router indicates that it is one of multiple Equal-Cost Multi-Path (ECMP) next hops for a given route. The asterisk will initially appear next to the first next hop for each route with multiple ECMP next hops. If the ARP entry for the *next hop** ages out or is cleared, then the next packet to be routed through the Brocade device whose destination matches that route can cause the asterisk to move to the next hop down the list of ECMP next hops for that route. This means that if the *next hop** goes down, the asterisk can move to another next hop with equal cost.
Port	The port through which this router sends packets to reach the route's destination.
Cost	The route's cost.
Type	The route type, which can be one of the following: • B – The route was learned from BGP. • D – The destination is directly connected to this Layer 3 Switch. • R – The route was learned from RIP. • S – The route is a static route. • * – The route and next-hop gateway are resolved through the **ip default-network** setting. • O – The route is an OSPF route. Unless you use the **ospf** option to display the route table, “O” is used for all OSPF routes. If you do use the **ospf** option, the following type codes are used: • O – OSPF intra area route (within the same area). • IA – The route is an OSPF inter area route (a route that passes from one area into another). • E1 – The route is an OSPF external type 1 route. • E2 – The route is an OSPF external type 2 route.

Clearing IP routes

If needed, you can clear the entire route table or specific individual routes.

To clear all routes from the IP route table, enter the following command.

```
Brocade# clear ip route
```

To clear route 10.157.22.0/24 from the IP routing table, enter the **clear ip route** command.

```
Brocade# clear ip route 10.157.22.0/24
```

Syntax: **clear ip route** [*ip-addr ip-mask*]

or

Syntax: **clear ip route** [*ip-addr*/*mask-bits*]

Displaying IP traffic statistics

To display IP traffic statistics, enter the **show ip traffic** command at any CLI level.

```
Brocade# show ip traffic
IP Statistics
  139 received, 145 sent, 0 forwarded
  0 filtered, 0 fragmented, 0 reassembled, 0 bad header
  0 no route, 0 unknown proto, 0 no buffer, 0 other errors

ICMP Statistics
Received:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation
Sent:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation

UDP Statistics
  1 received, 0 sent, 1 no port, 0 input errors

TCP Statistics
  0 active opens, 0 passive opens, 0 failed attempts
  0 active resets, 0 passive resets, 0 input errors
  138 in segments, 141 out segments, 4 retransmission

RIP Statistics
  0 requests sent, 0 requests received
  0 responses sent, 0 responses received
  0 unrecognized, 0 bad version, 0 bad addr family, 0 bad req format
  0 bad metrics, 0 bad resp format, 0 resp not from rip port
  0 resp from loopback, 0 packets rejected
```

The **show ip traffic** command displays the following information.

TABLE 22 CLI display of IP traffic statistics – Layer 3 Switch

Field	Description
IP statistics	
received	The total number of IP packets received by the device.
sent	The total number of IP packets originated and sent by the device.
forwarded	The total number of IP packets received by the device and forwarded to other devices.
filtered	The total number of IP packets filtered by the device.
fragmented	The total number of IP packets fragmented by this device to accommodate the MTU of this device or of another device.
reassembled	The total number of fragmented IP packets that this device re-assembled.
bad header	The number of IP packets dropped by the device due to a bad packet header.
no route	The number of packets dropped by the device because there was no route.
unknown proto	The number of packets dropped by the device because the value in the Protocol field of the packet header is unrecognized by this device.
no buffer	This information is used by Brocade customer support.
other errors	The number of packets dropped due to error types other than those listed above.
ICMP statistics The ICMP statistics are derived from RFC 792, “Internet Control Message Protocol”, RFC 950, “Internet Standard Subnetting Procedure”, and RFC 1256, “ICMP Router Discovery Messages”. Statistics are organized into Sent and Received. The field descriptions below apply to each.	
total	The total number of ICMP messages sent or received by the device.
errors	This information is used by Brocade customer support.
unreachable	The number of Destination Unreachable messages sent or received by the device.
time exceed	The number of Time Exceeded messages sent or received by the device.
parameter	The number of Parameter Problem messages sent or received by the device.
source quench	The number of Source Quench messages sent or received by the device.
redirect	The number of Redirect messages sent or received by the device.
echo	The number of Echo messages sent or received by the device.
echo reply	The number of Echo Reply messages sent or received by the device.
timestamp	The number of Timestamp messages sent or received by the device.
timestamp reply	The number of Timestamp Reply messages sent or received by the device.
addr mask	The number of Address Mask Request messages sent or received by the device.
addr mask reply	The number of Address Mask Replies messages sent or received by the device.
irdp advertisement	The number of ICMP Router Discovery Protocol (IRDP) Advertisement messages sent or received by the device.
irdp solicitation	The number of IRDP Solicitation messages sent or received by the device.
UDP statistics	

TABLE 22 CLI display of IP traffic statistics – Layer 3 Switch (Continued)

Field	Description
received	The number of UDP packets received by the device.
sent	The number of UDP packets sent by the device.
no port	The number of UDP packets dropped because they did not have a valid UDP port number.
input errors	This information is used by Brocade customer support.
TCP statistics The TCP statistics are derived from RFC 793, “Transmission Control Protocol”.	
active opens	The number of TCP connections opened by sending a TCP SYN to another device.
passive opens	The number of TCP connections opened by this device in response to connection requests (TCP SYNs) received from other devices.
failed attempts	This information is used by Brocade customer support.
active resets	The number of TCP connections this device reset by sending a TCP RESET message to the device at the other end of the connection.
passive resets	The number of TCP connections this device reset because the device at the other end of the connection sent a TCP RESET message.
input errors	This information is used by Brocade customer support.
in segments	The number of TCP segments received by the device.
out segments	The number of TCP segments sent by the device.
retransmission	The number of segments that this device retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment.
RIP statistics The RIP statistics are derived from RFC 1058, “Routing Information Protocol”.	
requests sent	The number of requests this device has sent to another RIP router for all or part of its RIP routing table.
requests received	The number of requests this device has received from another RIP router for all or part of this device RIP routing table.
responses sent	The number of responses this device has sent to another RIP router request for all or part of this device RIP routing table.
responses received	The number of responses this device has received to requests for all or part of another RIP router routing table.
unrecognized	This information is used by Brocade customer support.
bad version	The number of RIP packets dropped by the device because the RIP version was either invalid or is not supported by this device.
bad addr family	The number of RIP packets dropped because the value in the Address Family Identifier field of the packet header was invalid.
bad req format	The number of RIP request packets this router dropped because the format was bad.
bad metrics	This information is used by Brocade customer support.
bad resp format	The number of responses to RIP request packets dropped because the format was bad.
resp not from rip port	This information is used by Brocade customer support.

TABLE 22 CLI display of IP traffic statistics – Layer 3 Switch (Continued)

Field	Description
resp from loopback	The number of RIP responses received from loopback interfaces.
packets rejected	This information is used by Brocade customer support.

Displaying IP information – Layer 2 Switches

You can display the following IP configuration information statistics on Layer 2 Switches:

- Global IP settings – refer to "Displaying global IP configuration information" on page 128.
- ARP entries – refer to "Displaying ARP entries" on page 129.
- IP traffic statistics – refer to "Displaying IP traffic statistics" on page 129.

Displaying global IP configuration information

To display the Layer 2 Switch IP address and default gateway, enter the **show ip** command.

```
Brocade# show ip

     Switch IP address: 192.168.1.2

           Subnet mask: 255.255.255.0

Default router address: 192.168.1.1
   TFTP server address: None
Configuration filename: None
        Image filename: None
```

Syntax: **show ip**

This display shows the following information.

TABLE 23 CLI display of global IP configuration information – Layer 2 Switch

Field	Description
IP configuration	
Switch IP address	The management IP address configured on the Layer 2 Switch. Specify this address for Telnet .
Subnet mask	The subnet mask for the management IP address.
Default router address	The address of the default gateway, if you specified one.
Most recent TFTP access	
TFTP server address	The IP address of the most-recently contacted TFTP server, if the switch has contacted a TFTP server since the last time the software was reloaded or the switch was rebooted.
Configuration filename	The name under which the Layer 2 Switch startup-config file was uploaded or downloaded during the most recent TFTP access.
Image filename	The name of the Layer 2 Switch flash image (system software file) that was uploaded or downloaded during the most recent TFTP access.

Displaying ARP entries

To display the entries the Layer 2 Switch has placed in its ARP cache, enter the **show arp** command from any level of the CLI.

```
Brocade# show arp

Total Arp Entries : 1, maximum capacity: 1000
No.
1      IP               Mac                 Port Age VlanId
       192.168.1.170    0000.0011.d042       7    0       1
```

Syntax: **show arp**

This display shows the following information.

TABLE 24 CLI display of ARP cache

Field	Description
Total ARP Entries	The number of entries in the ARP cache.
Maximum capacity	The total number of ARP entries supported on the device.
IP	The IP address of the device.
Mac	The MAC address of the device. NOTE: If the MAC address is all zeros, the entry is for the default gateway, but the Layer 2 Switch does not have a link to the gateway.
Port	The port on which the entry was learned.
Age	The number of minutes the entry has remained unused. If this value reaches the ARP aging period, the entry is removed from the cache.
VlanId	The VLAN the port that learned the entry is in. NOTE: If the MAC address is all zeros, this field shows a random VLAN ID, since the Layer 2 Switch does not yet know which port the device for this entry is attached to.

Displaying IP traffic statistics

To display IP traffic statistics on a Layer 2 Switch, enter the **show ip traffic** command at any CLI level.

```
Brocade# show ip traffic

IP Statistics
  27 received, 24 sent
  0 fragmented, 0 reassembled, 0 bad header
  0 no route, 0 unknown proto, 0 no buffer, 0 other errors

ICMP Statistics
Received:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp rely, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation
Sent:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp rely, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation

UDP Statistics
  0 received, 0 sent, 0 no port, 0 input errors

TCP Statistics
  1 current active tcbs, 4 tcbs allocated, 0 tcbs freed 0 tcbs protected
  0 active opens, 0 passive opens, 0 failed attempts
  0 active resets, 0 passive resets, 0 input errors
 27 in segments, 24 out segments, 0 retransmission
```

Syntax: show ip traffic

The **show ip traffic** command displays the following information.

TABLE 25 CLI display of IP traffic statistics – Layer 2 Switch

Field	Description
IP statistics	
received	The total number of IP packets received by the device.
sent	The total number of IP packets originated and sent by the device.
fragmented	The total number of IP packets fragmented by this device to accommodate the MTU of this device or of another device.
reassembled	The total number of fragmented IP packets that this device re-assembled.
bad header	The number of IP packets dropped by the device due to a bad packet header.
no route	The number of packets dropped by the device because there was no route.
unknown proto	The number of packets dropped by the device because the value in the Protocol field of the packet header is unrecognized by this device.
no buffer	This information is used by Brocade customer support.
other errors	The number of packets that this device dropped due to error types other than the types listed above.
ICMP statistics The ICMP statistics are derived from RFC 792, “Internet Control Message Protocol”, RFC 950, “Internet Standard Subnetting Procedure”, and RFC 1256, “ICMP Router Discovery Messages”. Statistics are organized into Sent and Received. The field descriptions below apply to each.	

TABLE 25 CLI display of IP traffic statistics – Layer 2 Switch (Continued)

Field	Description
total	The total number of ICMP messages sent or received by the device.
errors	This information is used by Brocade customer support.
unreachable	The number of Destination Unreachable messages sent or received by the device.
time exceed	The number of Time Exceeded messages sent or received by the device.
parameter	The number of Parameter Problem messages sent or received by the device.
source quench	The number of Source Quench messages sent or received by the device.
redirect	The number of Redirect messages sent or received by the device.
echo	The number of Echo messages sent or received by the device.
echo reply	The number of Echo Reply messages sent or received by the device.
timestamp	The number of Timestamp messages sent or received by the device.
timestamp reply	The number of Timestamp Reply messages sent or received by the device.
addr mask	The number of Address Mask Request messages sent or received by the device.
addr mask reply	The number of Address Mask Replies messages sent or received by the device.
irdp advertisement	The number of ICMP Router Discovery Protocol (IRDP) Advertisement messages sent or received by the device.
irdp solicitation	The number of IRDP Solicitation messages sent or received by the device.
UDP statistics	
received	The number of UDP packets received by the device.
sent	The number of UDP packets sent by the device.
no port	The number of UDP packets dropped because the packet did not contain a valid UDP port number.
input errors	This information is used by Brocade customer support.
TCP statistics The TCP statistics are derived from RFC 793, “Transmission Control Protocol”.	
current active tcbs	The number of TCP Control Blocks (TCBs) that are currently active.
tcbs allocated	The number of TCBs that have been allocated.
tcbs freed	The number of TCBs that have been freed.
tcbs protected	This information is used by Brocade customer support.
active opens	The number of TCP connections opened by this device by sending a TCP SYN to another device.
passive opens	The number of TCP connections opened by this device in response to connection requests (TCP SYNs) received from other devices.
failed attempts	This information is used by Brocade customer support.
active resets	The number of TCP connections this device reset by sending a TCP RESET message to the device at the other end of the connection.
passive resets	The number of TCP connections this device reset because the device at the other end of the connection sent a TCP RESET message.
input errors	This information is used by Brocade customer support.

TABLE 25 CLI display of IP traffic statistics – Layer 2 Switch (Continued)

Field	Description
in segments	The number of TCP segments received by the device.
out segments	The number of TCP segments sent by the device.
retransmission	The number of segments that this device retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment.

Chapter

Base Layer 3 and Routing Protocols 2

Table 26 lists the base Layer 3 features Brocade ICX 6650 devices support. These features are supported in full Layer 3 software images, except where explicitly noted.

TABLE 26 Supported base Layer 3 features

Feature	Brocade ICX 6650
Static IP routing	Yes
Layer 3 system parameter limits	Yes
Static ARP entries	Yes
RIP V1 and V2 (Static RIP support only in the base Layer 3 image. The Brocade device with base Layer 3 does not learn RIP routes from other Layer 3 devices. However, the device does advertise directly connected routes.)	Yes
Redistribution of IP static routes into RIP	Yes
RIP default route learning	Yes
Route loop prevention: • Split horizon • Poison reverse	Yes
Route-only support (supported with full Layer 3 image only)	Yes

Adding a static IP route

To add a static IP route, enter a command such as the following at the global CONFIG level of the CLI.

```
Brocade(config)#ip route 192.168.10.0 255.255.255.0 192.168.2.1
```

This command adds a static IP route to the 192.168.10.x/24 subnet.

Syntax: [**no**] **ip route** *dest-ip-addr dest-mask next-hop-ip-addr* [*metric*] [**tag** *num*]

or

Syntax: [**no**] **ip route** *dest-ip-addr/mask-bits next-hop-ip-addr* [*metric*] [**tag** *num*]

The *dest-ip-addr* variable is the route destination. The *dest-mask* variable is the network mask for the route destination IP address. Alternatively, you can specify the network mask information by entering a forward slash followed by the number of bits in the network mask. For example, you can enter 192.168.0.0 255.255.255.0 as 192.168.0.0/.24. To configure a default route, enter 0.0.0.0 for *dest-ip-addr* and 0.0.0.0 for *dest-mask* (or 0 for the *mask-bits* variable if you specify the address in CIDR format). Specify the IP address of the default gateway using the *next-hop-ip-addr* variable.

The *next-hop-ip-addr* variable is the IP address of the next hop router (gateway) for the route.

The *metric* variable specifies the cost of the route and can be a number from 1 through 16. The default is 1. The metric is used by RIP. If you do not enable RIP, the metric is not used.

The **tag** *num* parameter specifies the tag value of the route. The possible value is from 0 through 4294967295. The default value is 0.

NOTE
You cannot specify **null0** or another interface as the next hop in the base Layer 3 image.

Adding a static ARP entry

Static entries are useful in cases where you want to pre-configure an entry for a device that is not connected to the Brocade device, or you want to prevent a particular entry from aging out. The software removes a dynamic entry from the ARP cache if the ARP aging interval expires before the entry is refreshed. Static entries do not age out, regardless of whether the Brocade device receives an ARP request from the device that has the entry address. The software places a static ARP entry into the ARP cache as soon as you create the entry.

To add a static ARP entry, enter a command such as the following at the global CONFIG level of the CLI.

```
Brocade(config)#arp 1 192.168.22.3 0000.00bb.cccc ethernet 1/1/3
```

This command adds a static ARP entry that maps IP address 192.168.22.3 to MAC address 0000.00bb.cccc. The entry is for a MAC address connected to Brocade port 3.

Syntax: [**no**] **arp** *num ip-addr mac-addr* **ethernet** *port*

The *num* variable specifies the entry number. You can specify a number from 1 up to the maximum number of static entries allowed on the device. You can allocate more memory to increase this amount. To do so, enter the **system-max ip-static-arp** *num* command at the global CONFIG level of the CLI.

The *ip-addr* variable specifies the IP address of the device that has the MAC address of the entry.

The *mac-addr* variable specifies the MAC address of the entry.

The **ethernet** *port* parameter specifies the port number attached to the device that has the MAC address of the entry. Specify the *port* variable in *stack-unit/slotnum/portnum* format

The **clear arp** command clears learned ARP entries but does not remove any static ARP entries.

Modifying and displaying Layer 3 system parameter limits

This section shows how to view and configure some of the Layer 3 system parameter limits.

Layer 3 configuration notes

Changing the system parameters reconfigures the device memory. Whenever you reconfigure the memory on a Brocade device, you must save the change to the startup-config file, and then reload the software to place the change into effect.

The Layer 3 system parameter limits for IPv6 models are automatically adjusted by the system and cannot be manually modified.

Displaying Layer 3 system parameter limits

To display the Layer 3 system parameter defaults, maximum values, and current values, enter the **show default value** command at any level of the CLI.

The following example shows the output on a Brocade ICX 6650 device.

```
Brocade#show default value
sys log buffers:50         mac age time:300 sec       telnet sessions:5

ip arp age:10 min          bootp relay max hops:4     ip ttl:64 hops
ip addr per intf:24

when multicast enabled :
igmp group memb.:260 sec   igmp query:125 sec         hardware drop: enabled

when ospf enabled :
ospf dead:40 sec           ospf hello:10 sec          ospf retrans:5 sec
ospf transit delay:1 sec

when bgp enabled :
bgp local pref.:100        bgp keep alive:60 sec      bgp hold:180 sec
bgp metric:10              bgp local as:1             bgp cluster id:0
bgp ext. distance:20       bgp int. distance:200      bgp local distance:200

System Parameters    Default    Maximum    Current
ip-arp               4000       64000      64000
ip-static-arp        512        6000       6000
multicast-route      64         8192       8192
```

Configuring RIP

If you want the Brocade device to use Routing Information Protocol (RIP), you must enable the protocol globally, and then enable RIP on individual ports. When you enable RIP on a port, you also must specify the version (version 1 only, version 2 only, or version 1 compatible with version 2).

Optionally, you also can set or change the following parameters:

- Route redistribution – You can enable the software to redistribute static routes from the IP route table into RIP. Redistribution is disabled by default.
- Learning of default routes – The default is disabled.
- Loop prevention (split horizon or poison reverse) – The default is poison reverse.

Enabling RIP

RIP is disabled by default. You must enable the protocol both globally and on the ports on which you want to use RIP.

To enable RIP globally, enter the following command.

```
Brocade(config)#router rip
```

Syntax: [**no**] **router rip**

To enable RIP on a port and specify the RIP version, enter commands such as the following.

```
Brocade(config-rip-router)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#ip rip v1-only
```

These commands change the CLI to the configuration level for port 1 and enable RIP version 1 on the interface. You must specify the version.

Syntax: **interface ethernet** *port*

Syntax: [**no**] **ip rip v1-only** | **v1-compatible-v2** | **v2-only**

Specify the *port* variable in the format *stack-unit/slotnum/portnum*.

Enabling redistribution of IP static routes into RIP

By default, the software does not redistribute the IP static routes in the route table into RIP. To configure redistribution, perform the following tasks.

1. Configure redistribution filters (optional).

   You can configure filters to permit or deny redistribution for a route based on the route metric. You also can configure a filter to change the metric. You can configure up to 64 redistribution filters. The software uses the filters in ascending numerical order and immediately takes the action specified by the filter. Thus, if filter 1 denies redistribution of a given route, the software does not redistribute the route, regardless of whether a filter with a higher ID permits redistribution of that route.

   NOTE
   The default redistribution action is permit, even after you configure and apply a permit or deny filter. To deny redistribution of specific routes, you must configure a deny filter.

NOTE
The option to set the metric is not applicable to static routes.

2. Enable redistribution.

NOTE
If you plan to configure redistribution filters, do not enable redistribution until you have configured the filters.

When you enable redistribution, all types of routes are redistributed into RIP; redistribution is not limited to IP static routes. If you want to deny certain routes from being redistributed into RIP, configure deny filters for those routes before you enable redistribution. You can configure up to 64 RIP redistribution filters. They are applied in ascending numerical order.

NOTE
The default redistribution action is permit, even after you configure and apply redistribution filters to the port. If you want to tightly control redistribution, apply a filter to deny all routes as the last filter (filter ID 64), and then apply filters with lower filter IDs to allow specific routes.

Configuring a redistribution filter

To configure a redistribution filter, enter a command such as the following.

```
Brocade(config-rip-router)#deny redistribute 1 static address 192.168.0.0
255.255.0.0
```

This command denies redistribution of all 192.168.x.x IP static routes.

Syntax: [**no**] **permit** | **deny redistribute** *filter-num* **static address** *ip-addr ip-mask*
[**match-metric** *value* | **set-metric** *value*]

The *filter-num* variable specifies the redistribution filter ID. Specify a number from 1 through 64. The software uses the filters in ascending numerical order. Thus, if filter 1 denies a route from being redistributed, the software does not redistribute that route even if a filter with a higher ID permits redistribution of the route.

The **static address** *ip-addr ip-mask* parameters apply redistribution to the specified network and subnet address. Use 0 to specify “any”. For example, “192.168.0.0 255.255.0.0“ means “any 192.168.x.x subnet”. However, to specify any subnet (all subnets match the filter), enter **static address 255.255.255.255 255.255.255.255**.

The **match-metric** *value* parameter applies redistribution to those routes with a specific metric value. Possible values are from 1 through 15.

The **set-metric** *value* parameter sets the RIP metric value that will be applied to the routes imported into RIP.

NOTE
The **set-metric** parameter does not apply to static routes.

The following command denies redistribution of a 192.168.x.x IP static route only if the route metric is 5.

```
Brocade(config-rip-router)#deny redistribute 2 static address 192.168.0.0
255.255.0.0 match-metric 5
```

The following commands deny redistribution of all routes except routes for 10.10.10.x and 10.20.20.x.

```
Brocade(config-rip-router)#deny redistribute 64 static address 255.255.255.255
255.255.255.255
Brocade(config-rip-router)#permit redistribute 1 static address 10.10.10.0
255.255.255.0
Brocade(config-rip-router)#permit redistribute 2 static address 10.20.20.0
255.255.255.0
```

Enabling redistribution

After you configure redistribution parameters, you must enable redistribution.

To enable RIP redistribution, enter the following command.

```
Brocade(config-rip-router)#redistribution
```

Syntax: [no] redistribution

Enabling learning of default routes

By default, the software does not learn RIP default routes.

To enable learning of default RIP routes, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#ip rip learn-default
```

Syntax: [no] ip rip learn-default

Changing the route loop prevention method

RIP can use the following methods to prevent routing loops:

- Split horizon – The Brocade device does not advertise a route on the same interface as the one on which it learned the route.
- Poison reverse – The Brocade device assigns a cost of 16 (“infinite” or “unreachable”) to a route before advertising it on the same interface as the one on which it learned the route. This is the default.

NOTE
These methods are in addition to the RIP maximum valid route cost of 15.

To enable split horizon, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#no ip rip poison-reverse
```

Syntax: [no] ip rip poison-reverse

Other Layer 3 protocols

For information about other IP configuration commands in the Layer 2 with base Layer 3 image that are not included in this chapter, refer to Chapter 1, “IP Configuration”.

For information about enabling or disabling Layer 3 routing protocols, refer to “Enabling or disabling routing protocols” on page 139.

Enabling or disabling routing protocols

This section describes how to enable or disable routing protocols. For complete configuration information about the routing protocols, refer to “RIP overview” on page 141.

The full Layer 3 code supports the following protocols:

- BGP4
- IGMP
- IP
- IP multicast (PIM-SM, PIM-DM)
- OSPF
- PIM
- RIPV1 and V2
- VRRP
- VRRP-E
- VSRP

IP routing is enabled by default on devices running Layer 3 code. All other protocols are disabled, so you must enable them to configure and use them.

To enable a protocol on a device running full Layer 3 code, enter **router** at the global CONFIG level, followed by the protocol to be enabled. The following example shows how to enable OSPF.

```
Brocade(config)#router ospf
```

Syntax: router bgp | igmp | ip | ospf | pim | rip |vrrp | vrrp-e | vsrp

Enabling or disabling Layer 2 switching

By default, Brocade Layer 3 switches support Layer 2 switching. These devices modify the routing protocols that are not supported on the devices. If you want to disable Layer 2 switching, you can do so globally or on individual ports, depending on the version of software your device is running.

NOTE
Consult your reseller or Brocade to understand the risks involved before disabling all Layer 2 switching operations.

Configuration notes and feature limitations for Layer 2 switching

- Enabling or disabling Layer 2 switching is supported in the full Layer 3 software image only.
- Enabling or disabling Layer 2 switching is not supported on virtual interfaces.

Command syntax for Layer 2 switching

To globally disable Layer 2 switching on a Layer 3 switch, enter commands such as the following.

```
Brocade(config)#route-only
Brocade(config)#exit
Brocade#write memory
Brocade#reload
```

To re-enable Layer 2 switching on a Layer 3 switch, enter the following commands.

```
Brocade(config)#no route-only
Brocade(config)#exit
Brocade#write memory
Brocade#reload
```

Syntax: [no] route-only

To disable Layer 2 switching only on a specific interface, go to the interface configuration level for that interface, and then disable the feature. The following commands show how to disable Layer 2 switching on port 2.

```
Brocade(config)#interface ethernet 1/1/2
Brocade(config-if-e10000-1/1/2)#route-only
```

Chapter

RIP (IPv4) 3

Table 27 lists the the Routing Information Protocol (RIP) for IPv4 features Brocade ICX 6650 devices support. These features are supported in the full Layer 3 software image.

TABLE 27 Supported RIP features

Feature	Brocade ICX 6650
RIP V1 and V2	Yes
Route learning and advertising	Yes
Route redistribution into RIP	Yes
Route metrics	Yes
Route loop prevention: • Poison reverse • Split horizon	Yes
RIP route advertisement suppression on a VRRP or VRRP-E backup interface	Yes
Route filters	Yes
CPU utilization statistics for RIP	Yes

RIP overview

Routing Information Protocol (RIP) is an IP route exchange protocol that uses a *distance vector* (a number representing a distance) to measure the cost of a given route. The *cost* is a distance vector because the cost often is equivalent to the number of router hops between the Brocade Layer 3 Switch and the destination network.

A Brocade Layer 3 Switch can receive multiple paths to a destination. The software evaluates the paths, selects the best path, and saves the path in the IP route table as the route to the destination. Typically, the best path is the path with the fewest hops. A hop is another router through which packets must travel to reach the destination. If the Brocade Layer 3 Switch receives a RIP update from another router that contains a path with fewer hops than the path stored in the Brocade Layer 3 Switch route table, the Layer 3 Switch replaces the older route with the newer one. The Layer 3 Switch then includes the new path in the updates it sends to other RIP routers, including Brocade Layer 3 Switches.

RIP routers, including the Brocade Layer 3 Switch, also can modify a route cost, generally by adding to it, to bias the selection of a route for a given destination. In this case, the actual number of router hops may be the same, but the route has an administratively higher cost and is thus less likely to be used than other, lower-cost routes.

A RIP route can have a maximum cost of 15. Any destination with a higher cost is considered unreachable. Although limiting to larger networks, the low maximum hop count prevents endless loops in the network.

Brocade Layer 3 Switches support the following RIP versions:

- Version (V1)
- V1 compatible with V2
- Version (V2) (the default)

RIP parameters and defaults

The following tables list the RIP parameters, their default values, and where to find configuration information.

RIP global parameters

Table 28 lists the global RIP parameters and their default values, and indicates where you can find configuration information.

TABLE 28 RIP global parameters

Parameter	Description	Default	Reference
RIP state	The global state of the protocol. **NOTE:** You also must enable the protocol on individual interfaces. Globally enabling the protocol does not allow interfaces to send and receive RIP information. Refer to Table 29 on page 143.	Disabled	page 144
Administrative distance	The administrative distance is a numeric value assigned to each type of route on the router. When the router is selecting from among multiple routes (sometimes of different origins) to the same destination, the router compares the administrative distances of the routes and selects the route with the lowest administrative distance. This parameter applies to routes originated by RIP. The administrative distance stays with a route when it is redistributed into other routing protocols.	120	page 146
Redistribution	RIP can redistribute routes from other routing protocols such as OSPF and BGP4 into RIP. A redistributed route is one that a router learns through another protocol, then distributes into RIP.	Disabled	page 146
Redistribution metric	RIP assigns a RIP metric (cost) to each external route redistributed from another routing protocol into RIP. An external route is a route with at least one hop (packets must travel through at least one other router to reach the destination). This parameter applies to routes that are redistributed from other protocols into RIP.	1 (one)	page 148
Update interval	How often the router sends route updates to its RIP neighbors.	30 seconds	page 149

TABLE 28 RIP global parameters (Continued)

Parameter	Description	Default	Reference
Learning default routes	The router can learn default routes from its RIP neighbors. NOTE: You also can enable or disable this parameter on an individual interface basis. Refer to Table 29 on page 143.	Disabled	page 149
Advertising and learning with specific neighbors	The Layer 3 Switch learns and advertises RIP routes with all its neighbors by default. You can prevent the Layer 3 Switch from advertising routes to specific neighbors or learning routes from specific neighbors.	Learning and advertising permitted for all neighbors	page 149

RIP interface parameters

Table 29 lists the interface-level RIP parameters and their default values, and indicates where you can find configuration information.

TABLE 29 RIP interface parameters

Parameter	Description	Default	Reference
RIP state and version	The state of the protocol and the version that is supported on the interface. The version can be one of the following: • Version 1 only • Version 2 only • Version 1, but also compatible with version 2 NOTE: You also must enable RIP globally.	Disabled	page 144
Metric	A numeric cost the router adds to RIP routes learned on the interface. This parameter applies only to RIP routes.	1 (one)	page 144
Learning default routes	Locally overrides the global setting. Refer to Table 28 on page 142.	Disabled	page 149
Loop prevention	The method a router uses to prevent routing loops caused by advertising a route on the same interface as the one on which the router learned the route. • Split horizon – The router does not advertise a route on the same interface as the one on which the router learned the route. • Poison reverse – The router assigns a cost of 16 (“infinite” or “unreachable”) to a route before advertising it on the same interface as the one on which the router learned the route.	Poison reverse NOTE: Enabling split horizon disables poison reverse on the interface.	page 150
Advertising and learning specific routes	You can control the routes that a Layer 3 Switch learns or advertises.	The Layer 3 Switch learns and advertises all RIP routes on all interfaces.	page 151

RIP parameter configuration

Use the following procedures to configure RIP parameters on a system-wide and individual interface basis.

Enabling RIP

RIP is disabled by default. To enable it, use the following procedure.

NOTE
You must enable the protocol globally and also on individual interfaces on which you want to advertise RIP. Globally enabling the protocol does not enable it on individual interfaces.

To enable RIP globally, enter the **router rip** command.

```
Brocade(config)#router rip
```

Syntax: [no] **router rip**

After globally enabling the protocol, you must enable it on individual interfaces. You can enable the protocol on physical interfaces as well as virtual routing interfaces. To enable RIP on an interface, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#ip rip v1-only
```

Syntax: [no] **ip rip v1-only** | **v1-compatible-v2** | **v2-only**

NOTE
You must specify the RIP version.

Enabling ECMP for routes in RIP

ECMP for routes in RIP is disabled by default. Use the **ecmp-enable** command to enable the feature at the router rip level.

```
Brocade(config-rip-router)#ecmp-enable
```

Syntax: [no] **ecmp-enable**

Configuring metric parameters

By default, a Brocade Layer 3 Switch port increases the cost of a RIP route that is learned on the port by one. You can configure individual ports to add more than one to a learned route cost. In addition, you can configure a RIP offset list to increase the metric for learned or advertised routes based on network address.

Changing the cost of routes learned on a port

By default, a Brocade Layer 3 Switch port increases the cost of a RIP route that is learned on the port. The Layer 3 Switch increases the cost by adding one to the route metric before storing the route.

You can change the amount that an individual port adds to the metric of RIP routes learned on the port. To do so, use the following method.

NOTE
RIP considers a route with a metric of 16 to be unreachable. Use this metric only if you do not want the route to be used. You can prevent the Layer 3 Switch from using a specific port for routes learned though that port by setting its metric to 16.

To increase the cost a port adds to RIP routes learned in that port, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/2
Brocade(config-if-e10000-1/1/2)#ip metric 5
```

These commands configure port 1/1/2 to add 5 to the cost of each route learned on the port.

Syntax: **ip metric** *metric-value*

The metric-value can be a value from 1 to 16.

Configuring a RIP offset list

A RIP offset list allows you to add to the metric of specific inbound or outbound routes learned or advertised by RIP. RIP offset lists provide a simple method for adding to the cost of specific routes and therefore biasing the Layer 3 Switch route selection away from those routes.

A RIP offset list consists of the following parameters:

- An access control list (ACL) that specifies the routes to which to add the metric.
- The direction:
  - In applies to routes the Layer 3 Switch learns from RIP neighbors.
  - Out applies to routes the Layer 3 Switch is advertising to its RIP neighbors.
- The type and number of a specific port to which the RIP offset list applies (optional).

The software adds the offset value to the routing metric (cost) of the routes that match the ACL. If a route matches both a global offset list and an interface-based offset list, the interface-based offset list takes precedence. The interface-based offset list metric is added to the route in this case.

You can configure up to 24 global RIP offset lists and up to 24 RIP offset lists on each interface.

To configure a global RIP offset list, enter commands such as the following.

```
Brocade(config)#access-list 21 deny 192.168.0.0 0.0.255.255
Brocade(config)#access-list 21 permit any
Brocade(config)#router rip
Brocade(config-rip-router)#offset-list 21 out 10
```

The commands in this example configure a standard ACL. The ACL matches on all IP networks except 192.168.x.x. When the Layer 3 Switch advertises a route that matches ACL 21, the offset list adds 10 to the route metric.

Syntax: [**no**] **offset-list** *ACL-number-or-name* **in** | **out** *metric* [**ethernet** *port*]

Specify *port* variable in the format *stack-unit/slotnum/portnum*.

In the following example, the Layer 3 Switch uses ACL 21 to add 10 to the metric of routes received on Ethernet port 1/1/2.

```
Brocade(config-rip-router)#offset-list 21 in 10 ethernet 1/1/2
```

Changing the administrative distance

By default, the Layer 3 Switch assigns the default RIP administrative distance (120) to RIP routes. When comparing routes based on administrative distance, the Layer 3 Switch selects the route with the lower distance. You can change the administrative distance for RIP routes.

NOTE
Refer to "Changing administrative distances" on page 313 for the default distances for all route sources.

To change the administrative distance for RIP routes, enter the **distance** command.

```
Brocade(config-rip-router)#distance 140
```

This command changes the administrative distance to 140 for all RIP routes.

Syntax: [**no**] **distance** *num*

The *num* variable specifies a range from 1 through 255.

Configuring redistribution

You can configure the Layer 3 Switch to redistribute routes learned through Open Shortest Path First (OSPF) or Border Gateway Protocol version 4 (BGP4) into RIP. When you redistribute a route from one of these other protocols into RIP, the Layer 3 Switch can use RIP to advertise the route to its RIP neighbors.

To configure redistribution, perform the following tasks:

1. Configure redistribution filters (optional). You can configure filters to permit or deny redistribution for a route based on its origin (OSPF, BGP4, and so on), the destination network address, and the route metric. You also can configure a filter to set the metric based on these criteria.
2. Change the default redistribution metric (optional). The Layer 3 Switch assigns a RIP metric of 1 to each redistributed route by default. You can change the default metric to a value up to 16.
3. Enable redistribution.

   NOTE
   Do not enable redistribution until you configure the other redistribution parameters.

Configuring redistribution filters

RIP redistribution filters apply to all interfaces. The software uses the filters in ascending numerical order and immediately takes the action specified by the filter. Thus, if filter 1 denies redistribution of a given route, the software does not redistribute the route, regardless of whether a filter with a higher ID would permit redistribution of that route.

NOTE
The default redistribution action is permit, even after you configure and apply redistribution filters to the virtual routing interface. If you want to tightly control redistribution, apply a filter to deny all routes as the last filter (the filter with the highest ID), and then apply filters with lower filter IDs to allow specific routes.

To configure a redistribution filter, enter a command such as the following.

```
Brocade(config-rip-router)#deny redistribute 2 all address 192.168.0.0
255.255.0.0
```

This command denies redistribution for all types of routes to the 192.168.x.x network.

Syntax: [**no**] **permit** | **deny redistribute** *filter-num* **all** | **bgp** | **ospf** | **static address** *ip-addr ip-mask* [**match-metric** *value* | **set-metric** *value*]

The *filter-num* variable specifies the redistribution filter ID. The software uses the filters in ascending numerical order. Thus, if filter 1 denies a route from being redistributed, the software does not redistribute that route even if a filter with a higher ID permits redistribution of the route.

The **all** parameter applies redistribution to all route types.

The **bgp** parameter applies redistribution to BGP4 routes only.

The **ospf** parameter applies redistribution to OSPF routes only.

The **static** parameter applies redistribution to IP static routes only.

The **address** *ip-addr ip-mask* parameters apply redistribution to the specified network and subnet address. Use 0 to specify “any”. For example, “192.168.0.0 255.255.0.0“ means “any 192.168.x.x subnet”. However, to specify any subnet (all subnets match the filter), enter “address 255.255.255.255 255.255.255.255”.

The **match-metric** *value* parameter applies the redistribution filter only to those routes with the specified metric value; possible values are from 1 through 15.

The **set-metric** *value* parameter sets the RIP metric value that will be applied to those routes imported into RIP.

The following command denies redistribution into RIP for all OSPF routes.

```
Brocade(config-rip-router)#deny redistribute 3 ospf address 192.168.0.0
255.255.0.0
```

The following command denies redistribution for all OSPF routes that have a metric of 10.

```
Brocade(config-rip-router)#deny redistribute 3 ospf address 192.168.0.0
255.255.0.0 match-metric 10
```

The following commands deny redistribution of all routes except routes for 10.10.10.x and 10.20.20.x.

```
Brocade(config-rip-router)#deny redistribute 64 static address 255.255.255.255
255.255.255.255
Brocade(config-rip-router)#permit redistribute 1 static address 10.10.10.0
255.255.255.0
Brocade(config-rip-router)#permit redistribute 2 static address 10.20.20.0
255.255.255.0
```

NOTE
This example assumes that the highest RIP redistribution filter ID configured on the device is 64.

Changing the redistribution metric

When the Layer 3 Switch redistributes a route into RIP, the software assigns a RIP metric (cost) to the route. By default, the software assigns a metric of 1 to each route that is redistributed into RIP. You can increase the metric that the Layer 3 Switch assigns up to 15.

To change the RIP metric the Layer 3 Switch assigns to redistributed routes, enter a command such as the following.

```
Brocade(config-rip-router)#default-metric 10
```

This command assigns a RIP metric of 10 to each route that is redistributed into RIP.

Syntax: [**no**] **default-metric** *metric-value*

The *metri-value* can be from 1 to 15.

Enabling redistribution

After you configure redistribution parameters, you need to enable redistribution.

To enable RIP redistribution, enter the **redistribution** command.

```
Brocade(config-rip-router)#redistribution
```

Syntax: [**no**] **redistribution**

The **no** form of this command disables RIP redistribution.

Removing a RIP redistribution deny filter

To remove a previously configured RIP redistribution deny filter, perform the following task:

1. Remove the RIP redistribution deny filter.
2. Disable the redistribution function.
3. Re-enable redistribution.

The following shows an example of how to remove a RIP redistribution deny filter.

```
Brocade(config-rip-router)#no deny redistribute 2 all address 192.168.0.0
255.255.0.0
Brocade(config-rip-router)#no redistribution
Brocade(config-rip-router)#redistribution
```

Route learning and advertising parameters

By default, a Brocade Layer 3 Switch learns routes from all its RIP neighbors and advertises RIP routes to those neighbors.

You can configure the following learning and advertising parameters:

- Update interval – The update interval specifies how often the Layer 3 Switch sends RIP route advertisements to its neighbors You can change the interval to a value from 1 through 1000 seconds. The default is 30 seconds.
- Learning and advertising of RIP default routes – The Layer 3 Switch learns and advertises RIP default routes by default. You can disable learning and advertising of default routes on a global or individual interface basis.
- Learning of standard RIP routes – By default, the Layer 3 Switch learns RIP routes from all its RIP neighbors. You can configure RIP neighbor filters to explicitly permit or deny learning from specific neighbors.

Changing the update interval for route advertisements

The update interval specifies how often the Layer 3 Switch sends route advertisements to its RIP neighbors. You can specify an interval from 1 through 1000 seconds. The default is 30 seconds.

To change the RIP update interval, enter a command such as the following.

```
Brocade(config-rip-router)#update-time 120
```

This command configures the Layer 3 Switch to send RIP updates every 120 seconds.

Syntax: **update-time** *interval*

The valid values for *interval* are from 1 to 1000.

Enabling learning of RIP default routes

You can enable learning of RIP default routes on a global or individual interface basis.

To enable learning of default RIP routes on a global basis, enter the following command.

```
Brocade(config-rip-router)#learn-default
```

Syntax: [**no**] **learn-default**

To enable learning of default RIP routes on an individual interface basis, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#ip rip learn-default
```

Syntax: [**no**] **ip rip learn-default**

Configuring a RIP neighbor filter

By default, a Brocade Layer 3 Switch learns RIP routes from all its RIP neighbors. Neighbor filters allow you to specify the neighbor routers from which the Brocade device can receive RIP routes. Neighbor filters apply globally to all ports.

To configure a RIP neighbor filter, enter a command such as the following.

```
Brocade(config-rip-router)#neighbor 1 deny any
```

This command configures the Layer 3 Switch so that the device does not learn any RIP routes from any RIP neighbors.

Syntax: [**no**] **neighbor** *filter-num* **permit** | **deny** *source-ip-address* | **any**

The following commands configure the Layer 3 Switch to learn routes from all neighbors except 192.168.1.170. Once you define a RIP neighbor filter, the default action changes from learning all routes from all neighbors to denying all routes from all neighbors except the ones you explicitly permit. To deny learning from a specific neighbor but allow all other neighbors, you must add a filter that allows learning from all neighbors. Be sure to add the filter to permit all neighbors last (the one with the highest filter number). Otherwise, the software can match on the permit all filter instead of a filter that denies a specific neighbor, and learn routes from that neighbor.

```
Brocade(config-rip-router)#neighbor 2 deny 192.168.1.170
Brocade(config-rip-router)#neighbor 1024 permit any
```

Denying route advertisements for connected routes

By default, RIP advertises all connected routes to neighboring routers except for the management route. To configure the router to not advertise connected routes, use the **dont-advertise-connected** command. When the **dont-advertise-connected** command is configured, the router only sends RIP enabled interface routes.

To configure the **dont-advertise-connected** command under the router RIP configuration level, enter the **router rip** command.

```
Brocade(config)#router rip

Brocade(config-rip-router)#dont-advertise-connected
```

Syntax: [no] dont-advertise-connected

To disable the configuration, use the **no** form of the command.

Changing the route loop prevention method

RIP can use the following methods to prevent routing loops:

- Split horizon – The Layer 3 Switch does not advertise a route on the same interface as the one on which the router learned the route.
- Poison reverse – The Layer 3 Switch assigns a cost of 16 (“infinite” or “unreachable”) to a route before advertising it on the same interface as the one on which the router learned the route. This is the default.

These loop prevention methods are configurable on an individual interface basis. One of the methods is always in effect on an interface enabled for RIP. If you disable one method, the other method is enabled.

NOTE
These methods may be used in addition to the RIP maximum valid route cost of 15.

Disabling poison reverse

To disable poison reverse and enable split horizon on an interface, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#no ip rip poison-reverse
```

Syntax: [no] ip rip poison-reverse

To disable split horizon and enable poison reverse on an interface, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/1
Brocade(config-if-e10000-1/1/1)#ip rip poison-reverse
```

Suppressing RIP route advertisement on a VRRP or VRRP-E backup interface

NOTE
This section applies only if you configure the Layer 3 Switch for Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E). Refer to Chapter 9, “VRRP and VRRP-E”.

Normally, a VRRP or VRRP-E backup includes route information for the virtual IP address (the backed-up interface) in RIP advertisements. As a result, other routers receive multiple paths for the backed-up interface and might sometimes unsuccessfully use the path to the backup rather than the path to the master.

You can prevent the backups from advertising route information for the backed-up interface by enabling suppression of the advertisements.

To suppress RIP advertisements for the backed-up interface, enter the following commands.

```
Brocade(config)#router rip
Brocade(config-rip-router)#use-vrrp-path
```

Syntax: [no] use-vrrp-path

The syntax is the same for VRRP and VRRP-E.

Configuring RIP route filters

You can configure RIP route filters to permit or deny learning or advertising of specific routes. Configure the filters globally, then apply them to individual interfaces. When you apply a RIP route filter to an interface, you specify whether the filter applies to learned routes (in) or advertised routes (out).

NOTE
A route is defined by the destination IP address and network mask.

NOTE
By default, routes that do not match a route filter are learned or advertised. To prevent a route from being learned or advertised, you must configure a filter to deny the route.

To configure RIP filters, enter commands such as the following.

```
Brocade(config-rip-router)#filter 1 permit 192.168.4.1 255.255.255.0
Brocade(config-rip-router)#filter 2 permit 192.168.5.1 255.255.255.0
Brocade(config-rip-router)#filter 3 permit 192.168.6.1 255.255.255.0
Brocade(config-rip-router)#filter 4 deny 192.168.7.1 255.255.255.0
```

These commands explicitly permit RIP routes to three networks, and deny the route to one network.

Because the default action is permit, all other routes (routes not explicitly permitted or denied by the filters) can be learned or advertised.

Syntax: **filter** *filter-num* **permit** | **deny** *source-ip-address* | **any** *source-mask* | **any** [**log**]

Applying a RIP route filter to an interface

Once you define RIP route filters, you must assign them to individual interfaces. The filters do not take effect until you apply them to interfaces. When you apply a RIP route filter, you also specify whether the filter applies to learned routes or advertised routes:

- Out filters apply to routes the Layer 3 Switch advertises to its neighbor on the interface.
- In filters apply to routes the Layer 3 Switch learns from its neighbor on the interface.

To apply RIP route filters to an interface, enter commands such as the following.

```
Brocade(config)#interface ethernet 1/1/2
Brocade(config-if-e10000-1/1/2)#ip rip filter-group in 2 3 4
```

These commands apply RIP route filters 2, 3, and 4 to all routes learned from the RIP neighbor on port 1/1/2.

Syntax: [**no**] **ip rip filter-group in** | **out** *filter-list*

Displaying RIP filters

To display the RIP filters configured on the router, enter the **show ip rip** command at any CLI level.

```
Brocade#show ip rip

                RIP Route Filter Table
  Index   Action   Route IP Address    Subnet Mask
  1       deny     any                 any
                RIP Neighbor Filter Table
  Index   Action   Neighbor IP Address
  1       permit   any
```

Syntax: **show ip rip**

Table 30 describes the information displayed by the **show ip rip** command.

TABLE 30 **CLI display of RIP filter information**

Field	Definition
Route filters The rows underneath "RIP Route Filter Table" list the RIP route filters. If no RIP route filters are configured on the device, the following message is displayed: "No Filters are configured in RIP Route Filter Table".	
Index	The filter number. You assign this number when you configure the filter.
Action	The action the router takes if a RIP route packet matches the IP address and subnet mask of the filter. The action can be one of the following: • deny – RIP route packets that match the address and network mask information in the filter are dropped. If applied to an interface outbound filter group, the filter prevents the router from advertising the route on that interface. If applied to an interface inbound filter group, the filter prevents the router from adding the route to its IP route table. • permit – RIP route packets that match the address and network mask information are accepted. If applied to an interface outbound filter group, the filter allows the router to advertise the route on that interface. If applied to an interface inbound filter group, the filter allows the router to add the route to its IP route table.
Route IP Address	The IP address of the route destination network or host.
Subnet Mask	The network mask for the IP address.
Neighbor filters The rows underneath "RIP Neighbor Filter Table" list the RIP neighbor filters. If no RIP neighbor filters are configured on the device, the following message is displayed: "No Filters are configured in RIP Neighbor Filter Table".	
Index	The filter number. You assign this number when you configure the filter.
Action	The action the router takes for RIP route packets to or from the specified neighbor: • deny – If the filter is applied to an interface outbound filter group, the filter prevents the router from advertising RIP routes to the specified neighbor on that interface. If the filter is applied to an interface inbound filter group, the filter prevents the router from receiving RIP updates from the specified neighbor. • permit – If the filter is applied to an interface outbound filter group, the filter allows the router to advertise RIP routes to the specified neighbor on that interface. If the filter is applied to an interface inbound filter group, the filter allows the router to receive RIP updates from the specified neighbor.
Neighbor IP Address	The IP address of the RIP neighbor.

Displaying CPU utilization statistics

You can display CPU utilization statistics for RIP and other IP protocols. To display CPU utilization statistics for RIP for the previous five-second, one-minute, five-minute, fifteen-minute, and runtime intervals, enter the **show process cpu** command at any level of the CLI.

```
Brocade#show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22            9
BGP              0.04      0.06      0.08      0.14            13
GVRP             0.00      0.00      0.00      0.00            0
ICMP             0.00      0.00      0.00      0.00            0
IP               0.00      0.00      0.00      0.00            0
OSPF             0.00      0.00      0.00      0.00            0
RIP              0.04      0.07      0.08      0.09            7
STP              0.00      0.00      0.00      0.00            0
VRRP             0.00      0.00      0.00      0.00            0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running, as shown in the following example.

```
Brocade#show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00            0
BGP              0.00      0.00      0.00      0.00            0
GVRP             0.00      0.00      0.00      0.00            0
ICMP             0.01      0.00      0.00      0.00            1
IP               0.00      0.00      0.00      0.00            0
OSPF             0.00      0.00      0.00      0.00            0
RIP              0.00      0.00      0.00      0.00            0
STP              0.00      0.00      0.00      0.00            0
VRRP             0.00      0.00      0.00      0.00            0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following.

```
Brocade#show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00        0
BGP              0.00        0
GVRP             0.00        0
ICMP             0.01        1
IP               0.00        0
OSPF             0.00        0
RIP              0.00        0
STP              0.01        0
VRRP             0.00        0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is for the previous 1 second and 80 milliseconds.

Syntax: **show process cpu** [*num*]

The *num* parameter specifies the number of seconds and can be from 1 through 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous five-second, one-minute, five-minute, and fifteen-minute intervals.

Chapter

RIP (IPv6)

4

Table 31 lists the Routing Information Protocol (RIP) for IPv6 features Brocade ICX 6650 devices support. These features are supported with premium IPv6 devices running the full Layer 3 software image

TABLE 31 **Supported RIPng features**

Feature	Brocade ICX 6650
RIPng	Yes
Up to 2K RIPng routes	Yes
RIPng timers	Yes
Route learning and advertising	Yes
Route redistribution into RIPng	Yes
Route loop prevention: • Poison reverse • Split horizon	Yes
Clearing RIPng routes	Yes

RIPng overview

Routing Information Protocol (RIP) is an IP route exchange protocol that uses a distance vector (a number representing a distance) to measure the cost of a given route. RIP uses a hop count as its cost or metric.

IPv6 RIP, known as *Routing Information Protocol Next Generation* or RIPng functions similarly to IPv4 RIP Version 2. RIPng supports IPv6 addresses and prefixes and introduces some new commands that are specific to RIPng. This chapter describes the commands that are specific to RIPng. This section does not describe commands that apply to both IPv4 RIP and RIPng.

RIPng maintains a *Routing Information Base (RIB),* which is a local route table. The local RIB contains the lowest-cost IPv6 routes learned from other RIP routers. In turn, RIPng attempts to add routes from its local RIB into the main IPv6 route table.

NOTE
Brocade ICX 6650 IPv6 devices support up to 2000 RIPng routes.

This section describes the following:

- How to configure RIPng
- How to clear RIPng information from the RIPng route table
- How to display RIPng information and statistics

Summary of configuration tasks

To configure RIPng, you must enable RIPng globally on the Brocade device and on individual router interfaces. The following configuration tasks are optional:

- Change the default settings of RIPng timers
- Configure how the Brocade device learns and advertises routes
- Configure which routes are redistributed into RIPng from other sources
- Configure how the Brocade device distributes routes through RIPng
- Configure poison reverse parameters

RIPng configuration

Before configuring the Brocade device to run RIPng, you must do the following:

- Enable the forwarding of IPv6 traffic on the Brocade device using the **ipv6 unicast-routing** command.
- Enable IPv6 on each interface on which you plan to enable RIPng. You enable IPv6 on an interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

Enabling RIPng

By default, RIPng is disabled. To enable RIPng, you must enable it globally on the Brocade device and also on individual router interfaces.

NOTE
You are required to configure a router ID when running only IPv6 routing protocols.

NOTE
Enabling RIPng globally on the Brocade device does not enable it on individual router interfaces.

To enable RIPng globally, enter the following command.

```
Brocade(config)#ipv6 router rip
Brocade(config-ripng-router)#
```

After you enter this command, the Brocade device enters the RIPng configuration level, where you can access several commands that allow you to configure RIPng.

Syntax: [**no**] **ipv6 router rip**

To disable RIPng globally, use the **no** form of this command.

After enabling RIPng globally, you must enable it on individual router interfaces. You can enable it on physical as well as virtual routing interfaces. For example, to enable RIPng on Ethernet interface 1/1/3, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)# ipv6 rip enable
```

Syntax: [**no**] **ipv6 rip enable**

To disable RIPng on an individual router interface, use the **no** form of this command.

RIPng timers

Table 32 describes the RIPng timers and provides their defaults.

TABLE 32 **RIPng timers**

Timer	Description	Default
Update	Amount of time (in seconds) between RIPng routing updates.	30 seconds.
Timeout	Amount of time (in seconds) after which a route is considered unreachable.	180 seconds.
Hold-down	Amount of time (in seconds) during which information about other paths is ignored.	180 seconds.
Garbage-collection	Amount of time (in seconds) after which a route is removed from the routing table.	120 seconds.

You can adjust these timers for RIPng. Before doing so, keep the following caveats in mind:

- If you adjust these RIPng timers, Brocade strongly recommends setting the same timer values for all routers and access servers in the network.
- Setting the update timer to a shorter interval can cause the routers to spend excessive time updating the IPv6 route table.
- Brocade recommends setting the timeout timer value to at least three times the value of the update timer.
- Brocade recommends a shorter hold-down timer interval, because a longer interval can cause delays in RIPng convergence.

Updating RIPng timers

The following example sets updates to be broadcast every 45 seconds. If a route is not heard from in 135 seconds, the route is declared unusable. Further information is suppressed for an additional 10 seconds. Assuming no updates, the route is flushed from the routing table 20 seconds after the end of the hold-down period.

```
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# timers 45 135 10 20
```

Syntax: [no] **timers** *update-timer timeout-timer hold-down-timer garbage-collection-timer*

Possible values for the timers are as follows

- Update timer: 3 through 65535 seconds
- Timeout timer: 9 through 65535 seconds
- Hold-down timer: 9 through 65535 seconds
- Garbage-collection timer: 9 through 65535 seconds

NOTE
You must enter a value for each timer, even if you want to retain the current setting of a particular timer.

To return to the default values of the RIPng timers, use the **no** form of this command.

Route learning and advertising parameters

You can configure the following learning and advertising parameters:

- Learning and advertising of RIPng default routes
- Advertising of IPv6 address summaries
- Metric of routes learned and advertised on a router interface

By default, the Brocade device does not learn IPv6 default routes (::/0). You can originate default routes into RIPng, which causes individual router interfaces to include the default routes in their updates. When configuring the origination of the default routes, you can also do the following:

- Suppress all other routes from the updates
- Include all other routes in the updates

Configuring default route learning and advertising

To originate default routes in RIPng and suppress all other routes in updates sent from Ethernet interface 1/1/4, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)# ipv6 rip default-information only
```

To originate IPv6 default routes and include all other routes in updates sent from Ethernet interface 1/1/4, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)# ipv6 rip default-information originate
```

Syntax: [no] **ipv6 rip default-information only** | **originate**

The **only** keyword originates the default routes and suppresses all other routes from the updates.

The **originate** keyword originates the default routes and includes all other routes in the updates.

To remove the explicit default routes from RIPng and suppress advertisement of these routes, use the **no** form of this command.

Advertising IPv6 address summaries

You can configure RIPng to advertise a summary of IPv6 addresses from a router interface and to specify an IPv6 prefix that summarizes the routes.

If a route prefix length matches the value specified in the **ipv6 rip summary-address** command, RIPng advertises the prefix specified in the **ipv6 rip summary-address** command instead of the original route.

For example, to advertise the summarized prefix 2001:DB8::/36 instead of the IPv6 address 2001:DB8:0:adff:8935:e838:78:e0ff with a prefix length of 64 bits from Ethernet interface 1/1/4, enter the following commands.

```
Brocade(config)# interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)# ipv6 address 2001:db8:0:adff:8935:e838:78:
e0ff /64
Brocade(config-if-e10000-1/1/4)# ipv6 rip summary-address 2001:db8::/36
```

Syntax: [no] **ipv6 rip summary-address** *ipv6-prefix/prefix-length*

You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

To stop the advertising of the summarized IPv6 prefix, use the **no** form of this command.

Changing the metric of routes learned and advertised on an interface

A router interface increases the metric of an incoming RIPng route it learns by an offset (the default is 1). The Brocade device then places the route in the route table. When the Brocade device sends an update, it advertises the route with the metric plus the default offset of zero in an outgoing update message.

You can change the metric offset an individual interface adds to a route learned by the interface or advertised by the interface. For example, to change the metric offset for incoming routes learned by Ethernet interface 1/1/4 to 1 and the metric offset for outgoing routes advertised by the interface to 3, enter the following commands.

```
Brocade(config)#interface ethernet 1/1/4
Brocade(config-if-e10000-1/1/4)#ipv6 rip metric-offset 1
Brocade(config-if-e10000-1/1/4)#ipv6 rip metric-offset out 3
```

In this example, if Ethernet interface 1/1/4 learns about an incoming route, it will increase the incoming metric by 2 (the default offset of 1 and the additional offset of 1 as specified in this example). If Ethernet interface 1/1/4 advertises an outgoing route, it will increase the metric by 3 as specified in this example.

Syntax: [**no**] **ipv6 rip metric-offset** *offset-value*

The valid value for *offset-value* are from 1 to 16.

Syntax: [**no**] **ipv6 rip metric-offset out** *outgoing-offset-valu*e

The valid value for *outgoing-offset-value* are from 1 to 15.

To return the metric offset to its default value, use the **no** form of this command.

Redistributing routes into RIPng

You can configure the Brocade device to redistribute routes from the following sources into RIPng:

- IPv6 static routes
- Directly connected IPv6 networks
- OSPF V3

When you redistribute a route from IPv6 or OSPF V3 into RIPng, the Brocade device can use RIPng to advertise the route to its RIPng neighbors.

When configuring the Brocade device to redistribute routes, you can optionally specify a metric for the redistributed routes. If you do not explicitly configure a metric, the default metric value of 1 is used.

For example, to redistribute OSPF V3 routes into RIPng, enter the following commands.

```
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# redistribute ospf
```

Syntax: **redistribute bgp** | **connected** | **isis** | **ospf** | **static** [**metric** *number*]

For the metric, specify a numerical value that is consistent with RIPng.

Controlling distribution of routes through RIPng

You can create a prefix list and then apply it to RIPng routing updates that are received or sent on a router interface. Performing this task allows you to control the distribution of routes through RIPng.

For example, to permit the inclusion of routes with the prefix 2001:DB8::/16 in RIPng routing updates sent from Ethernet interface 1/1/4, enter the following commands.

```
Brocade(config)# ipv6 prefix-list routesfor2001 permit 2001:db8::/16
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# distribute-list prefix-list routesfor2001 out
ethernet 1/1/4
```

To deny prefix lengths greater than 64 bits in routes that have the prefix 2001:DB8::/64 and allow all other routes received on tunnel interface 1, enter the following commands.

```
Brocade(config)# ipv6 prefix-list 3ee0routes deny 2001:db8::/64 le 128
Brocade(config)# ipv6 prefix-list 3ee0routes permit ::/0 ge 1 le 128
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# distribute-list prefix-list 3ee0routes in
tunnel 1
```

Syntax: [**no**] **distribute-list prefix-list** *name* **in** | **out** *interface port*

The *name* parameter indicates the name of the prefix list generated using the **ipv6 prefix-list** command.

The **in** keyword indicates that the prefix list is applied to incoming routing updates on the specified interface.

The **out** keyword indicates that the prefix list is applied to outgoing routing updates on the specified interface.

For the *interface* parameter, you can specify the **ethernet**, **loopback**, **ve**, or **tunnel** keywords. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE or tunnel interface, also specify the VE or tunnel number.

To remove the prefix list, use the **no** form of this command.

Configuring poison reverse parameters

By default, poison reverse is disabled on a RIPng router. If poison reverse is enabled, RIPng advertises routes it learns from a particular interface over that same interface with a metric of 16, which means that the route is unreachable.

If poison reverse is enabled on the RIPng router, it takes precedence over split horizon (if it is also enabled).

To enable poison reverse on the RIPng router, enter the following commands.

```
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# poison-reverse
```

Syntax: [**no**] **poison-reverse**

To disable poison reverse, use the **no** form of this command.

By default, if a RIPng interface goes down, the Brocade device does not send a triggered update for the interface IPv6 networks.

To better handle this situation, you can configure a RIPng router to send a triggered update containing the local routes of the disabled interface with an unreachable metric of 16 to the other RIPng routers in the routing domain. You can enable the sending of a triggered update by entering the following commands.

```
Brocade(config)# ipv6 router rip
Brocade(config-ripng-router)# poison-local-routes
```

Syntax: [**no**] **poison-local-routes**

To disable the sending of a triggered update, use the **no** form of this command.

Clearing RIPng routes from the IPv6 route table

To clear all RIPng routes from the RIPng route table and the IPv6 main route table and reset the routes, enter the following command at the Privileged EXEC level or any of the CONFIG levels of the CLI.

```
Brocade# clear ipv6 rip route
```

Syntax: **clear ipv6 rip route**

Displaying the RIPng configuration

To display RIPng configuration information, enter the **show ipv6 rip** command at any CLI level.

```
Brocade# show ipv6 rip
 IPv6 rip enabled, port 521
     Administrative distance is 120
     Updates every 30 seconds, expire after 180
     Holddown lasts 180 seconds, garbage collect after 120
     Split horizon is on; poison reverse is off
     Default routes are not generated
     Periodic updates 0, trigger updates 0
     Distribute List, Inbound : Not set
     Distribute List, Outbound : Not set
     Redistribute: CONNECTED
```

Syntax: **show ipv6 rip**

Table 33 describes the information displayed by the **show ipv6 rip** command.

TABLE 33 RIPng configuration fields

Field	Description
IPv6 RIP status/port	The status of RIPng on the Brocade device. Possible status is “enabled” or “disabled.” The UDP port number over which RIPng is enabled.
Administrative distance	The setting of the administrative distance for RIPng.
Updates/expiration	The settings of the RIPng update and timeout timers.
Holddown/garbage collection	The settings of the RIPng hold-down and garbage-collection timers.
Split horizon/poison reverse	The status of the RIPng split horizon and poison reverse features. Possible status is “on” or “off.”
Default routes	The status of RIPng default routes.
Periodic updates/trigger updates	The number of periodic updates and triggered updates sent by the RIPng router.
Distribution lists	The inbound and outbound distribution lists applied to RIPng.
Redistribution	The types of IPv6 routes redistributed into RIPng. The types can include the following: • STATIC – IPv6 static routes are redistributed into RIPng. • CONNECTED – Directly connected IPv6 networks are redistributed into RIPng. • OSPF – OSPF V3 routes are redistributed into RIPng.

Displaying RIPng routing table

To display the RIPng routing table, enter the **show ipv6 rip route** command at any CLI level.

```
Brocade# show ipv6 rip route
IPv6 RIP Routing Table - 4 entries:
 2001:db8::/64, from 2001:db8:212:f2ff:fe87:9a40, ve 177    (314)
          RIP, metric 2, tag 0, timers: aging 11
 2001:db8:11::/64, from ::, null   (0)
          CONNECTED, metric 1, tag 0, timers: none
 2001:db8:2001:102:1:1::/96, from ::, ve 102    (1)
          LOCAL, metric 1, tag 0, timers: none
 2001:db8:2061:31:1:1:1::/96, from 2001:db8:2e0:52ff:fe88:8000, ve 100    (58)
          RIP, metric 2, tag 0, timers: aging 43
```

Syntax: **show ipv6 rip route** [*ipv6-prefix/prefix-length* | *ipv6-address*]

The *ipv6-prefix/prefix-length* parameters restrict the display to the entries for the specified IPv6 prefix. You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

The *ipv6-address* parameter restricts the display to the entries for the specified IPv6 address. You must specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

Table 34 describes the information displayed by the **show ipv6 rip route** command.

TABLE 34 **RIPng routing table fields**

Field	Description
IPv6 RIPng Routing Table entries	The total number of entries in the RIPng routing table.
ipv6-prefix/prefix-length OR ipv6-address	The IPv6 prefix and prefix length. The IPv6 address.
Next-hop router	The next-hop router for this Brocade device. If :: appears, the route is originated locally.
Interface	The interface name. If “null” appears, the interface is originated locally.
Serial number	The number identifying the order in which the route was added to the RIPng route table.
Source of route	The source of the route information. The source can be one of the following: • LOCAL – Routes configured on local interfaces taking part in RIPng. • RIP – Routes learned by RIPng. • CONNECTED – IPv6 routes redistributed from directly connected networks. • STATIC – IPv6 static routes are redistributed into RIPng. • OSPF – OSPF V3 routes are redistributed into RIPng.
Metric *number*	The cost of the route. The *number* parameter indicates the number of hops to the destination.
Tag *number*	The tag value of the route.
Timers:	Indicates if the hold-down timer or the garbage-collection timer is set.

Chapter

OSPF version 2 (IPv4) 5

Table 35 lists the Open Shortest Path First (OSPF) Version 2 (IPv4) features Brocade ICX 6650 devices support. These features are supported in the full Layer 3 software image only.

TABLE 35 Supported OSPF V2 features

Feature	Brocade ICX 6650
OSPF V2	Yes
OSPF point-to-point links	Yes
RFC 1583 and RFC 2178 compliant	Yes
Support for OSPF RFC 2328 Appendix E	Yes
Dynamic OSPF activation and configuration	Yes
Dynamic OSPFmemory	Yes
OSPF graceful restart	Yes
Assigning OSPF V2 areas	Yes
Assigning interfaces to an area	Yes
Timer for OSPF authentication changes	Yes
Block flooding of outbound LSAs on specific interfaces	Yes
OSPF non-broadcast interface	Yes
Virtual links	Yes
Changing the reference bandwidth for the cost on OSPF interfaces	Yes
Route redistribution filters	Yes
Prevent specific OSPF routes from being installed in the IP route table	Yes
Load sharing	Yes
Configuring default route origination	Yes
SPF timers	Yes
Modifying redistribution metric type	Yes
Modifying administrative distance	Yes
OSPF group LSA pacing	Yes
OSPF traps	Yes
Exit overflow interval	Yes
Syslog messages	Yes
Clearing OSPF information	Yes

This chapter describes how to configure OSPF Version 2 on Brocade Layer 3 Switches using the CLI. OSPF Version 2 is supported on devices running IPv4.

NOTE
The terms *Layer 3 Switch* and *router* are used interchangeably in this chapter and mean the same thing.

OSPF overview

Open Shortest Path First (OSPF) is a link-state routing protocol. The protocol uses link-state advertisements (LSAs) to update neighboring routers regarding its interfaces and information on those interfaces. The router floods these LSAs to all neighboring routers to update them regarding the interfaces. Each router maintains an identical database that describes its area topology to help a router determine the shortest path between it and any neighboring router.

Brocade Layer 3 Switches support the following types of LSAs, which are described in RFC 1583:

- Router link
- Network link
- Summary link
- Autonomous system (AS) summary link
- AS external link
- Not-So-Stubby Area (NSSA) external link
- Grace LSAs

OSPF is built upon a hierarchy of network components. The highest level of the hierarchy is the Autonomous System (AS). An autonomous system is defined as a number of networks, all of which share the same routing and administration characteristics.

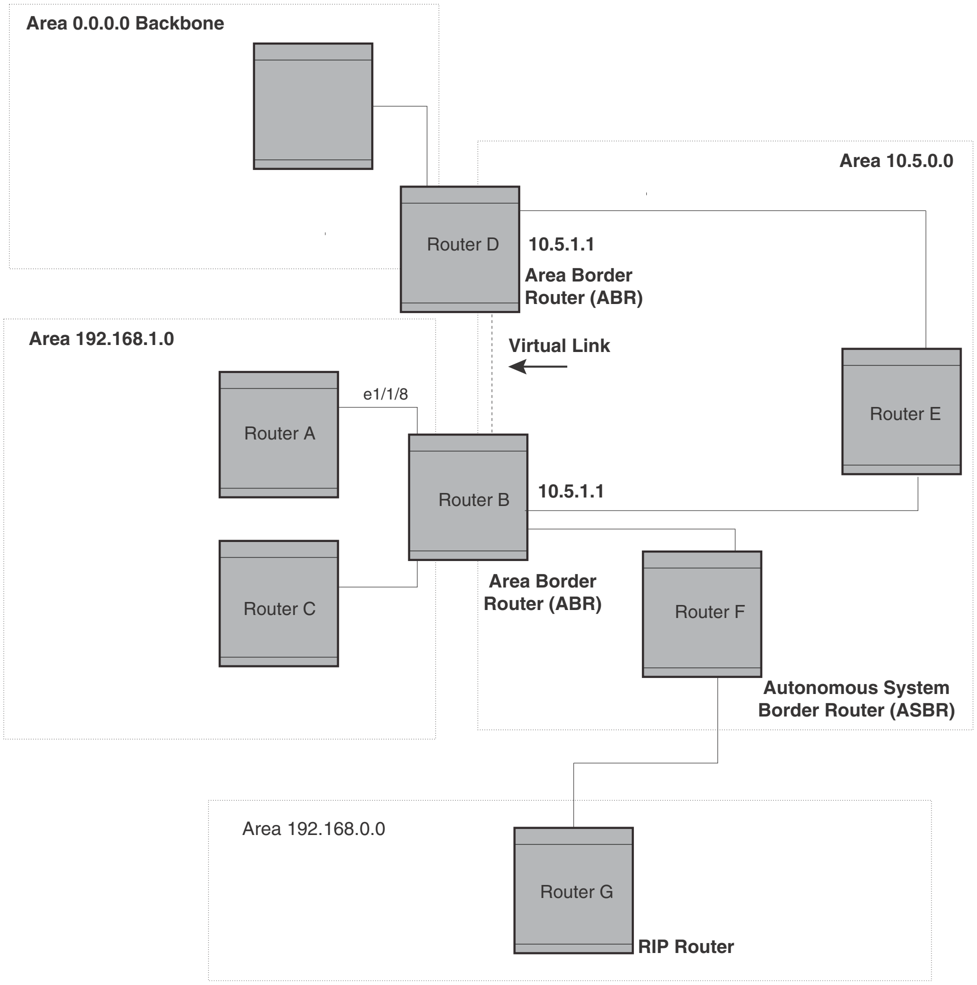
An AS can be divided into multiple areas as shown in Figure 17 on page 169. Each area represents a collection of contiguous networks and hosts. Areas limit the area to which link-state advertisements are broadcast, thereby limiting the amount of flooding that occurs within the network. An area is represented in OSPF by either an IP address or a number.

You can further limit the broadcast area of flooding by defining an area range. The area range allows you to assign an aggregate value to a range of IP addresses. This aggregate value becomes the address that is advertised instead all of the individual addresses it represents being advertised. You can assign up to 32 ranges in an OSPF area.

An OSPF router can be a member of multiple areas. Routers with membership in multiple areas are known as Area Border Routers (ABRs). Each ABR maintains a separate topological database for each area the router is in. Each topological database contains all of the LSA databases for each router within a given area. The routers within the same area have identical topological databases. The ABR is responsible for forwarding routing information or changes between its border areas.

An Autonomous System Boundary Router (ASBR) is a router that is running multiple protocols and serves as a gateway to routers outside an area and those operating with different protocols. The ASBR is able to import and translate different protocol routes into OSPF through a process known as *redistribution*. For more details on redistribution and configuration examples, refer to "Enabling route redistribution" on page 200.

FIGURE 17 **OSPF operating in a network**



OSPF point-to-point links

One important OSPF process is ***Adjacency***. Adjacency occurs when a relationship is formed between neighboring routers for the purpose of exchanging routing information. Adjacent OSPF neighbor routers go beyond the simple Hello packet exchange; they exchange database information. In order to minimize the amount of information exchanged on a particular segment, one of the first steps in creating adjacency is to assign a Designated Router (DR) and a Backup Designated Router (BDR). The Designated Router ensures that there is a central point of contact, thereby improving convergence time within a multi-access segment.

In an OSPF point-to-point network, where a direct Layer 3 connection exists between a single pair of OSPF routers, there is no need for Designated and Backup Designated Routers, as is the case in OSPF multi-access networks. Without the need for Designated and Backup Designated routers, a point-to-point network establishes adjacency and converges faster. The neighboring routers become adjacent whenever they can communicate directly. In contrast, in broadcast and non-broadcast multi-access (NBMA) networks, the Designated Router and Backup Designated Router become adjacent to all other routers attached to the network.

To configure an OSPF point-to-point link, refer to "Configuring an OSPF point-to-point link" on page 211.

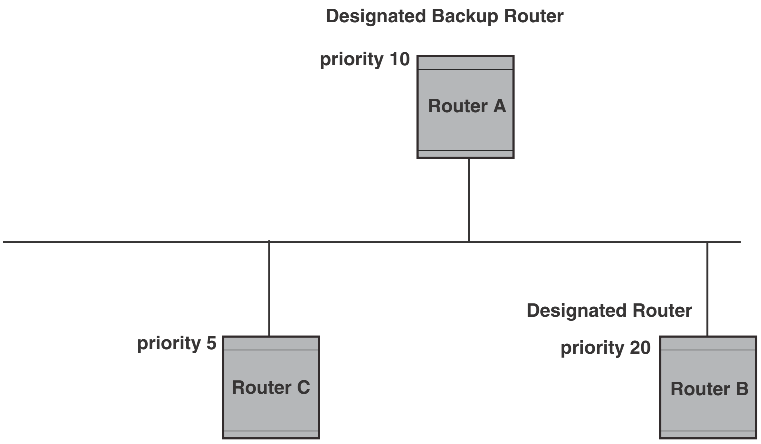
Designated routers in multi-access networks

In a network that has multiple routers attached, OSPF elects one router to serve as the designated router (DR) and another router on the segment to act as the backup designated router (BDR). This arrangement minimizes the amount of repetitive information that is forwarded on the network by forwarding all messages to the designated router and backup designated routers responsible for forwarding the updates throughout the network.

Designated router election in multi-access networks

In a network with no designated router and no backup designated router, the neighboring router with the highest priority is elected as the DR, and the router with the next largest priority is elected as the BDR, as shown in Figure 18

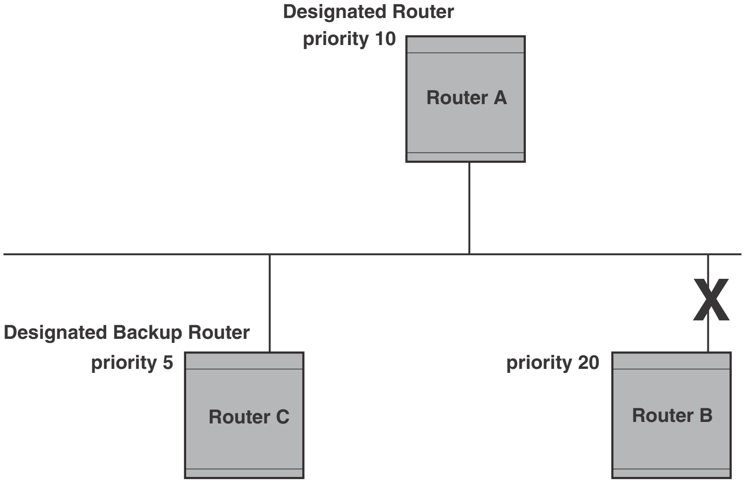
FIGURE 18 Designated and backup router election



If the DR goes off-line, the BDR automatically becomes the DR. The router with the next highest priority becomes the new BDR. This process is shown in Figure 19.

NOTE
Priority is a configurable option at the interface level. You can use this parameter to help bias one router as the DR.

FIGURE 19 **Backup designated router becomes designated router**



If two neighbors share the same priority, the router with the highest router ID is designated as the DR. The router with the next highest router ID is designated as the BDR.

NOTE
By default, the Brocade router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, refer to "Changing the router ID" on page 31.

When multiple routers on the same network are declaring themselves as DRs, then both priority and router ID are used to select the designated router and backup designated routers.

When only one router on the network claims the DR role despite neighboring routers with higher priorities or router IDs, this router remains the DR. This is also true for BDRs.

The DR and BDR election process is performed when one of the following events occurs:

- An interface is in a waiting state and the wait time expires
- An interface is in a waiting state and a hello packet is received that addresses the BDR
- A change in the neighbor state occurs, such as:
  - A neighbor state transitions from 2 or higher
  - Communication to a neighbor is lost
  - A neighbor declares itself to be the DR or BDR for the first time

OSPF RFC 1583 and 2178 compliance

Brocade routers are configured, by default, to be compliant with the RFC 1583 OSPF V2 specification. Brocade routers can also be configured to operate with the latest OSPF standard, RFC 2178.

NOTE
For details on how to configure the system to operate with the RFC 2178, refer to "Modifying the OSPF standard compliance setting" on page 210.

Reduction of equivalent AS External LSAs

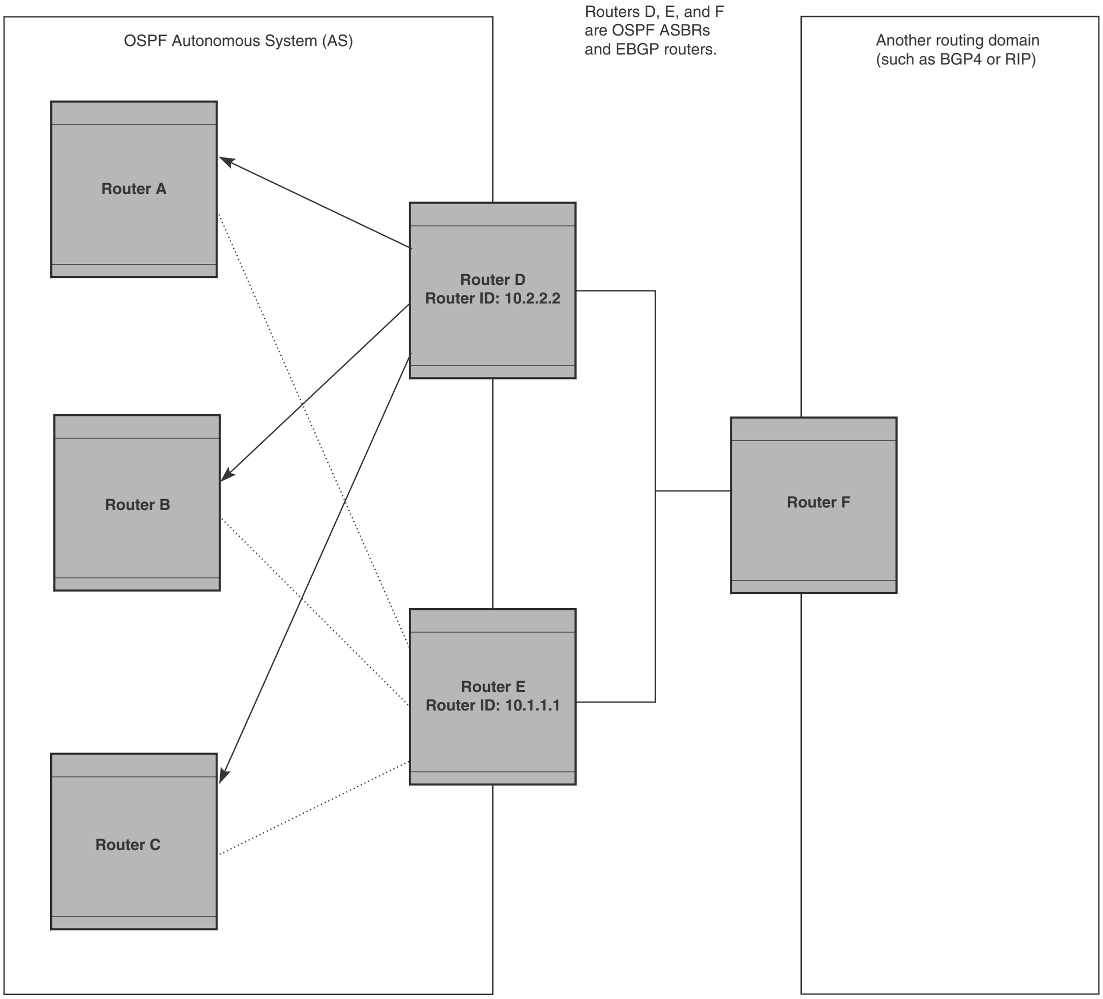
An OSPF ASBR uses AS External link advertisements (AS External LSAs) to originate advertisements of a route to another routing domain, such as a BGP4 or RIP domain. The ASBR advertises the route to the external domain by flooding AS External LSAs to all the other OSPF routers (except those inside stub networks) within the local OSPF Autonomous System (AS).

In some cases, multiple ASBRs in an AS can originate equivalent LSAs. The LSAs are equivalent when they have the same cost, the same next hop, and the same destination. Brocade devices optimize OSPF by eliminating duplicate AS External LSAs in this case. The Layer 3 Switch with the lower router ID flushes the duplicate External LSAs from its database and thus does not flood the duplicate External LSAs into the OSPF AS. AS External LSA reduction therefore reduces the size of the Layer 3 Switch link state database.

This enhancement implements the portion of RFC 2328 that describes AS External LSA reduction. This enhancement is enabled by default, requires no configuration, and cannot be disabled.

Figure 20 shows an example of the AS External LSA reduction feature. In this example, Brocade Layer 3 Switches D and E are OSPF ASBRs, and thus communicate route information between the OSPF AS, which contains Routers A, B, and C, and another routing domain, which contains Router F. The other routing domain is running another routing protocol, such as BGP4 or RIP. Routers D, E, and F, therefore, are each running both OSPF and either BGP4 or RIP.

FIGURE 20 AS External LSA reduction



Notice that both Router D and Router E have a route to the other routing domain through Router F. In earlier software releases, if Routers D and E have equal-cost routes to Router F, then both Router D and Router E flood AS External LSAs to Routers A, B, and C advertising the route to Router F. Since both routers are flooding equivalent routes, Routers A, B, and C receive multiple routes with the same cost to the same destination (Router F). For Routers A, B, and C, either route to Router F (through Router D or through Router E) is equally good.

OSPF eliminates the duplicate AS External LSAs. When two or more Brocade Layer 3 Switches configured as ASBRs have equal-cost routes to the same next-hop router in an external routing domain, the ASBR with the highest router ID floods the AS External LSAs for the external domain into the OSPF AS, while the other ASBRs flush the equivalent AS External LSAs from their databases. As a result, the overall volume of route advertisement traffic within the AS is reduced and the Layer 3 Switches that flush the duplicate AS External LSAs have more memory for other OSPF data. In Figure 20, since Router D has a higher router ID than Router E, Router D floods the AS External LSAs for Router F to Routers A, B, and C. Router E flushes the equivalent AS External LSAs from its database.

Algorithm for AS External LSA reduction

Figure 20 shows an example in which the normal AS External LSA reduction feature is in effect. The behavior changes under the following conditions:

- There is one ASBR advertising (originating) a route to the external destination, but one of the following happens:
  - A second ASBR comes on-line
  - A second ASBR that is already on-line begins advertising an equivalent route to the same destination.

  In either case above, the router with the higher router ID floods the AS External LSAs and the other router flushes its equivalent AS External LSAs. For example, if Router D is offline, Router E is the only source for a route to the external routing domain. When Router D comes on-line, it takes over flooding of the AS External LSAs to Router F, while Router E flushes its equivalent AS External LSAs to Router F.
- One of the ASBRs starts advertising a route that is no longer equivalent to the route the other ASBR is advertising. In this case, the ASBRs each flood AS External LSAs. Since the LSAs either no longer have the same cost or no longer have the same next-hop router, the LSAs are no longer equivalent, and the LSA reduction feature no longer applies.
- The ASBR with the higher router ID becomes unavailable or is reconfigured so that it is no longer an ASBR. In this case, the other ASBR floods the AS External LSAs. For example, if Router D goes off-line, then Router E starts flooding the AS with AS External LSAs for the route to Router F.

Support for OSPF RFC 2328 Appendix E

Brocade devices provide support for Appendix E in OSPF RFC 2328. Appendix E describes a method to ensure that an OSPF router (such as aBrocade Layer 3 Switch) generates unique link state IDs for type-5 (External) link state advertisements (LSAs) in cases where two networks have the same network address but different network masks.

NOTE
Support for Appendix E of RFC 2328 is enabled automatically and cannot be disabled. No user configuration is required.

Normally, an OSPF router uses the network address alone for the link state ID of the link state advertisement (LSA) for the network. For example, if the router needs to generate an LSA for network 10.1.2.3 255.0.0.0, the router generates ID 10.1.2.3 for the LSA.

However, suppose that an OSPF router needs to generate LSAs for all the following networks:

- 10.0.0.0 255.0.0.0
- 10.0.0.0 255.255.0.0
- 10.0.0.0 255.255.255.0

All three networks have the same network address, 10.0.0.0. Without support for RFC 2328 Appendix E, an OSPF router uses the same link state ID, 10.0.0.0, for the LSAs for all three networks. For example, if the router generates an LSA with ID 10.0.0.0 for network 10.0.0.0 255.0.0.0, this LSA conflicts with the LSA generated for network 10.0.0.0 255.255.0.0 or 10.0.0.0 255.255.255.0. The result is multiple LSAs that have the same ID but that contain different route information.

When Appendix E is supported, the router generates the link state ID for a network as follows.

1. Does an LSA with the network address as its ID already exist?
   - No – Use the network address as the ID.
   - Yes – Go to step 2.
2. Compare the networks that have the same network address, to determine which network is more specific. The more specific network is the one that has more contiguous one bits in its network mask. For example, network 10.0.0.0 255.255.0.0 is more specific than network 10.0.0.0 255.0.0.0, because the first network has 16 ones bits (255.255.0.0) whereas the second network has only 8 ones bits (255.0.0.0):
   - For the less specific network, use the networks address as the ID.
   - For the more specific network, use the network broadcast address as the ID. The broadcast address is the network address, with all ones bits in the host portion of the address. For example, the broadcast address for network 10.0.0.0 255.255.0.0 is 10.0.255.255.

   If this comparison results in a change to the ID of an LSA that has already been generated, the router generates a new LSA to replace the previous one. For example, if the router has already generated an LSA for network with ID 10.0.0.0 for network 10.0.0.0 255.255.255.0, the router must generate a new LSA for the network, if the router needs to generate an LSA for network 10.0.0.0 255.255.0.0 or 10.0.0.0 255.0.0.0.

Dynamic OSPF activation and configuration

OSPF is automatically activated when you enable it. The protocol does not require a software reload.

You can configure and save the following OSPF changes without resetting the system:

- All OSPF interface-related parameters (for example: area, hello timer, router dead time cost, priority, re-transmission time, transit delay)
- All area parameters
- All area range parameters
- All virtual-link parameters
- All global parameters
- Creation and deletion of an area, interface or virtual link

In addition, you can make the following changes without a system reset by first disabling and then re-enabling OSPF operation:

- Changes to address ranges
- Changes to global values for redistribution
- Addition of new virtual links

You also can change the amount of memory allocated to various types of LSA entries. However, these changes require a system reset or reboot.

Dynamic OSPF memory

Brocade ICX 6650 devices dynamically allocate memory for Link State Advertisements (LSAs) and other OSPF data structures. This eliminates overflow conditions and does not require a reload to change OSPF memory allocation. So long as the Layer 3 Switch has free (unallocated) dynamic memory, OSPF can use the memory.

To display the current allocations of dynamic memory, use the **show memory** command.

OSPF graceful restart

OSPF graceful restart is a high-availability routing feature that minimizes disruption in traffic forwarding, diminishes route flapping, and provides continuous service during a system restart, including restart events that occur during a switchover, failover OS upgrade. During such events, routes remain available between devices.

When OSPF graceful restart is enabled, a restarting router sends special LSAs, called grace LSAs, to its neighbors either before a planned OSPF restart or immediately after an unplanned restart. The grace LSAs specify a grace period for neighbors of the restarting router to continue using the existing routes to and through the router after a restart. When the restarting router comes back up, it continues to use its existing OSPF routes as if nothing happened. In the background, the router relearns its neighbors prior to the restart, recalculates its OSPF routes, and replaces existing routes with new routes as necessary. Once the grace period has passed, adjacent routers resume normal operation.

OSPF graceful restart is enabled globally by default. In this configuration, all OSPF neighbors are subject to the graceful restart capability. Neighbor routers must support the helper mode of OSPF graceful restart, which is enabled by default on all FastIron Layer 3 switches.

NOTE
If a Brocade ICX 6650 device is configured for OSPF graceful restart and is intended to be used in switchover, the OSPF dead-interval should be changed to 60 seconds on OSPF interfaces to ensure that the graceful restart process succeeds without a timeout. Instructions for changing the OSPF dead-interval are provided in “Modifying interface defaults” on page 184.

The Brocade implementation of OSPF graceful restart supports RFC 3623: Graceful OSPF Restart.

For details on how to configure OSPF graceful restart, refer to “Configuring OSPF graceful restart” on page 211.

Configuring OSPF

Perform the following steps to begin using OSPF on the router.

1. "Enabling OSPF on the router" on page 178
2. "Assigning OSPF areas" on page 179
3. "Assigning an area range (optional)" on page 183
4. "Assigning interfaces to an area" on page 184.
5. "Defining redistribution filters" on page 194
6. "Enabling route redistribution" on page 200.
7. "Modifying the OSPF standard compliance setting" on page 210

NOTE
OSPF is automatically enabled without a system reset.

OSPF configuration rules

- Brocade ICX 6650 devices support a maximum of 676 OSPF interfaces.
- If a router is to operate as an ASBR, you must enable the ASBR capability at the system level.
- Redistribution must be enabled on routers configured to operate as ASBRs.
- All router ports must be assigned to one of the defined areas on an OSPF router. When a port is assigned to an area, all corresponding subnets on that port are automatically included in the assignment.

OSPF parameters

You can modify or set the following global and interface OSPF parameters.

Global parameters

- Modify the OSPF standard compliance setting
- Assign OSPF areas
- Define an area range
- Define the area virtual link
- Set global default metric for OSPF
- Change the reference bandwidth for the default cost of OSPF interfaces
- Disable or re-enable load sharing
- Enable or disable default-information-originate
- Modify Shortest Path First (SPF) timers
- Define external route summarization
- Modify the redistribution metric type
- Define deny redistribution
- Define permit redistribution

- Enable redistribution
- Change the LSA pacing interval
- Modify OSPF Traps generated
- Modify database overflow interval

Interface parameters

- Assign interfaces to an area
- Define the authentication key for the interface
- Change the authentication-change interval
- Modify the cost for a link
- Modify the dead interval
- Modify MD5 authentication key parameters
- Modify the priority of the interface
- Modify the retransmit interval for the interface
- Modify the transit delay of the interface

NOTE
When using the CLI, you set global level parameters at the OSPF CONFIG level of the CLI. To reach that level, enter **router ospf…** at the global CONFIG level. Interface parameters for OSPF are set at the interface CONFIG level using the CLI command, **ip ospf…**

Enabling OSPF on the router

When you enable OSPF on the router, the protocol is automatically activated. To enable OSPF on the router, enter the following CLI command.

```
Brocade(config)#router ospf
```

This command launches you into the OSPF router level where you can assign areas and modify OSPF global parameters.

Syntax: **router ospf**

Note regarding disabling OSPF

If you disable OSPF, the Layer 3 Switch removes all the configuration information for the disabled protocol from the running-config. Moreover, when you save the configuration to the startup-config file after disabling one of these protocols, all the configuration information for the disabled protocol is removed from the startup-config file.

NOTE
If you do not want to delete the OSPF configuration information, use the CLI command **clear ip ospf all** instead of **no router ospf**. Refer to "Resetting OSPF" on page 179.

When you enter the **no router ospf** command, the CLI displays a warning message such as the following.

```
Brocade(config-ospf-router)#no router ospf
router ospf mode now disabled. All ospf config data will be lost when writing to
flash!
```

If you have disabled the protocol but have not yet saved the configuration to the startup-config file and reloaded the software, you can restore the configuration information by re-entering the command to enable the protocol (for example, **router ospf**). If you have already saved the configuration to the startup-config file and reloaded the software, the information is gone.

If you are testing an OSPF configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

Resetting OSPF

The **clear ip ospf all** command globally resets (disables then re-enables) OSPF without deleting the OSPF configuration information. This command is equivalent to entering the commands **no router ospf** followed by **router ospf.** Whereas the **no router ospf** command disables OSPF and removes all the configuration information for the disabled protocol from the running-config, the **router ospf** command re-enables OSPF and restores the OSPF configuration information.

The **clear ip ospf all** command is useful If you are testing an OSPF configuration and are likely to disable and re-enable the protocol. This way, you do not have to save the configuration after disabling the protocol, and you do not have to restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

To reset OSPF without deleting the OSPF configuration, enter the following command at the Global CONFIG level or at the Router OSPF level of the CLI.

```
Brocade#clear ip ospf all
```

Syntax: **clear ip ospf all**

Assigning OSPF areas

Once OSPF is enabled on the system, you can assign areas. Assign an IP address or number as the area ID for each area. The area ID is representative of all IP addresses (subnets) on a router port. Each port on a router can support one area.

An area can be normal, a stub, or a Not-So-Stubby Area (NSSA):

- Normal – OSPF routers within a normal area can send and receive External Link State Advertisements (LSAs).
- Stub – OSPF routers within a stub area cannot send or receive External LSAs. In addition, OSPF routers in a stub area must use a default route to the area Area Border Router (ABR) or Autonomous System Boundary Router (ASBR) to send traffic out of the area.

- NSSA – The ASBR of an NSSA can import external route information into the area:
  - ASBRs redistribute (import) external routes into the NSSA as type 7 LSAs. Type-7 External LSAs are a special type of LSA generated only by ASBRs within an NSSA, and are flooded to all the routers within only that NSSA.
  - ABRs translate type 7 LSAs into type 5 External LSAs, which can then be flooded throughout the AS. You can configure address ranges on the ABR of an NSSA so that the ABR converts multiple type-7 External LSAs received from the NSSA into a single type-5 External LSA.

    When an NSSA contains more than one ABR, OSPF elects one of the ABRs to perform the LSA translation for NSSA. OSPF elects the ABR with the highest router ID. If the elected ABR becomes unavailable, OSPF automatically elects the ABR with the next highest router ID to take over translation of LSAs for the NSSA. The election process for NSSA ABRs is automatic.

Example

To set up the OSPF areas shown in Figure 17 on page 169, enter the following commands.

```
Brocade(config-ospf-router)#area 192.168.1.0
Brocade(config-ospf-router)#area 10.5.0.0
Brocade(config-ospf-router)#area 192.168.0.0
Brocade(config-ospf-router)#area 0.0.0.0
Brocade(config-ospf-router)#write memory
```

Syntax: **area** *num* | *ip-addr*

The *num* | *ip-addr* parameter specifies the area number, which can be a number or in IP address format. If you specify a number, the number can be from 0 through 18.

NOTE
You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

Assigning a totally stubby area

By default, the Layer 3 Switch sends summary LSAs (LSA type 3) into stub areas. You can further reduce the number of link state advertisements (LSAs) sent into a stub area by configuring the Layer 3 Switch to stop sending summary LSAs (type 3 LSAs) into the area. You can disable the summary LSAs when you are configuring the stub area or later after you have configured the area.

This feature disables origination of summary LSAs, but the Layer 3 Switch still accepts summary LSAs from OSPF neighbors and floods them to other neighbors. The Layer 3 Switch can form adjacencies with other routers regardless of whether summarization is enabled or disabled for areas on each router.

When you enter a command to disable the summary LSAs, the change takes effect immediately. If you apply the option to a previously configured area, the Layer 3 Switch flushes all of the summary LSAs it has generated (as an ABR) from the area.

NOTE
This feature applies only when the Layer 3 Switch is configured as an Area Border Router (ABR) for the area. To completely prevent summary LSAs from being sent to the area, disable the summary LSAs on each OSPF router that is an ABR for the area.

This feature does not apply to Not-So-Stubby Areas (NSSAs).

To disable summary LSAs for a stub area, enter commands such as the following.

```
Brocade(config-ospf-router)#area 40 stub 99 no-summary
```

Syntax: **area** *num* | *ip-addr* **stub** *cost* [**no-summary**]

The *num* | *ip-addr* parameter specifies the area number, which can be a number or in IP address format. If you specify a number, the number can be from 0 through 18.

The **stub** *cost* parameter specifies an additional cost for using a route to or from this area and can be from 1 through 16777215. There is no default. Normal areas do not use the cost parameter.

The **no-summary** parameter applies only to stub areas and disables summary LSAs from being sent into the area.

NOTE
You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

Assigning a Not-So-Stubby Area

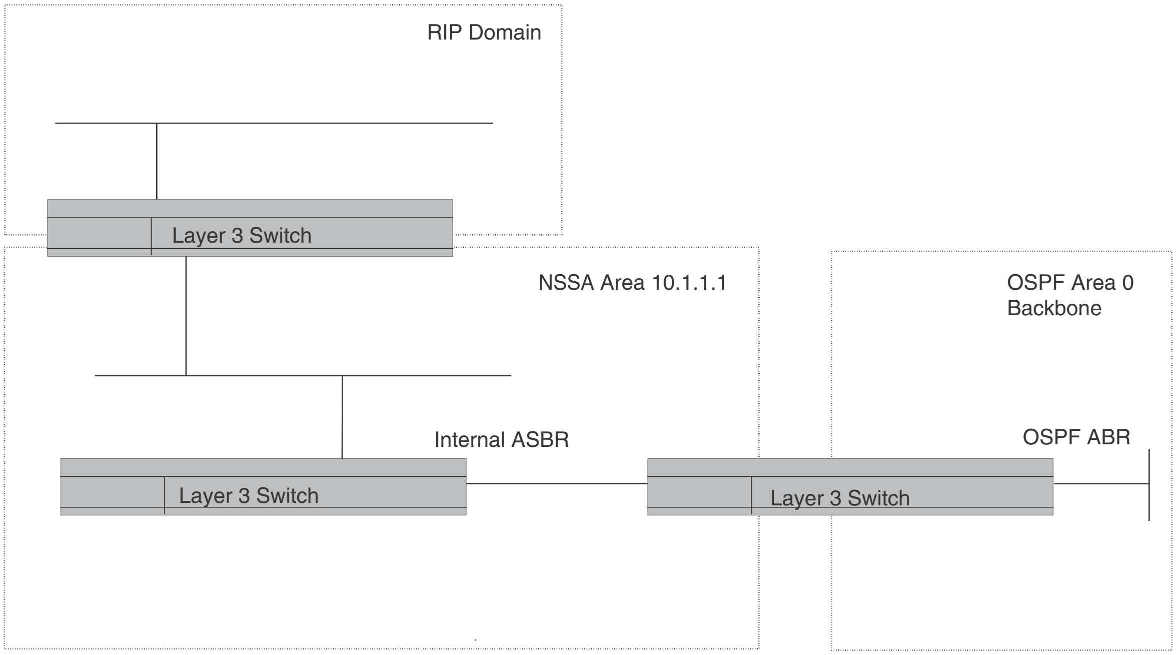
The OSPF Not-So-Stubby Area (NSSA) feature enables you to configure OSPF areas that provide the benefits of stub areas, but that also are capable of importing external route information. OSPF does not flood external routes from other areas into an NSSA, but does translate and flood route information from the NSSA into other areas such as the backbone.

NSSAs are especially useful when you want to summarize Type-5 External LSAs (external routes) before forwarding them into an OSPF area. The OSPF specification (RFC 2328) prohibits summarization of Type-5 LSAs and requires OSPF to flood Type-5 LSAs throughout a routing domain. When you configure an NSSA, you can specify an address range for aggregating the external routes that the NSSA's ABR exports into other areas.

The Brocade implementation of NSSA is based on RFC 1587.

Figure 21 shows an example of an OSPF network containing an NSSA.

FIGURE 21 OSPF network containing an NSSA



This example shows two routing domains, a RIP domain and an OSPF domain. The ASBR inside the NSSA imports external routes from RIP into the NSSA as Type-7 LSAs, which the ASBR floods throughout the NSSA.

The ABR translates the Type-7 LSAs into Type-5 LSAs. If an area range is configured for the NSSA, the ABR also summarizes the LSAs into an aggregate LSA before flooding the Type-5 LSAs into the backbone.

Since the NSSA is partially “stubby” the ABR does not flood external LSAs from the backbone into the NSSA. To provide access to the rest of the Autonomous System (AS), the ABR generates a default Type-7 LSA into the NSSA.

Configuring an NSSA

To configure OSPF area 10.1.1.1 as an NSSA, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#area 10.1.1.1 nssa 1
Brocade(config-ospf-router)#write memory
```

Syntax: **area** *num* | *ip-addr* **nssa** *cost* | **default-information-originate**

The *num* | *ip-addr* parameter specifies the area number, which can be a number or in IP address format. If you specify a number, the number can be from 0 through 18.

The **nssa** *cost* | **default-information-originate** parameter specifies that this is a Not-So-Stubby Area (NSSA). The *cost* specifies an additional cost for using a route to or from this NSSA and can be from 1 through 16777215. There is no default. Normal areas do not use the cost parameter. Alternatively, the **default-information-originate** parameter causes the Layer 3 Switch to inject the default route into the NSSA.

NOTE
The Layer 3 Switch does not inject the default route into an NSSA by default.

NOTE
You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

To configure additional parameters for OSPF interfaces in the NSSA, use the **ip ospf area...** command at the interface level of the CLI.

Configuring a summary address for the NSSA

If you want the ABR that connects the NSSA to other areas to summarize the routes in the NSSA before translating them into Type-5 LSAs and flooding them into the other areas, configure a summary address. The ABR creates an aggregate value based on the summary address. The aggregate value becomes the address that the ABR advertises instead of advertising the individual addresses represented by the aggregate.

To configure a summary address in NSSA 10.1.1.1, enter the following commands. This example assumes that you have already configured NSSA 10.1.1.1.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#summary-address 192.168.22.1 255.255.0.0
Brocade(config-ospf-router)#write memory
```

Syntax: [**no**] **summary-address** *ip-addr ip-mask*

The *ip-addr* parameter specifies the IP address portion of the range. The software compares the address with the significant bits in the mask. All network addresses that match this comparison are summarized in a single route advertised by the router.

The *ip-mask* parameter specifies the portions of the IP address that a route must contain to be summarized in the summary route. In the example above, all networks that begin with 192.168 are summarized into a single route.

Assigning an area range (optional)

You can assign a range for an area, but it is not required. Ranges allow a specific IP address and mask to represent a range of IP addresses within an area, so that only that reference range address is advertised to the network, instead of all the addresses within that range. Each area can have up to 32 range addresses.

Example

To define an area range for subnets on 192.168.5.1 and 192.168.6.2, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#area 192.168.5.1 range 192.168.0.0 255.255.0.0
Brocade(config-ospf-router)#area 192.168.6.2 range 192.168.0.0 255.255.0.0
```

Syntax: **area** *num* | *ip-addr* **range** *ip-addr ip-mask*

The *num* | *ip-addr* parameter specifies the area number, which can be in IP address format.

The **range** *ip-addr* parameter specifies the IP address portion of the range. The software compares the address with the significant bits in the mask. All network addresses that match this comparison are summarized in a single route advertised by the router.

The *ip-mask* parameter specifies the portions of the IP address that a route must contain to be summarized in the summary route. In the example above, all networks that begin with 193.45 are summarized into a single route.

Assigning interfaces to an area

Once you define OSPF areas, you can assign interfaces to the areas. All router ports must be assigned to one of the defined areas on an OSPF router. When a port is assigned to an area, all corresponding subnets on that port are automatically included in the assignment.

To assign interface 1/1/8 to area 192.168.0.0 and then save the changes, enter the following commands.

```
Brocade(config-ospf-router)#interface e 1/1/8
Brocade(config-if-e10000-1/1/8)#ip ospf area 192.168.0.0
Brocade(config-if-e10000-1/1/8)#write memory
```

Modifying interface defaults

OSPF has interface parameters that you can configure. For simplicity, each of these parameters has a default value. No change to these default values is required except as needed for specific network configurations.

Port default values can be modified using the following commands at the interface configuration level of the CLI:

- **ip ospf area** *ip-addr*
- **ip ospf auth-change-wait-time** *secs*
- **ip ospf authentication-key** [0 | 1] *string*
- **ip ospf cost** *num*
- **ip ospf dead-interval** *value*
- **ip ospf hello-interval** *value*
- **ip ospf md5-authentication key-activation-wait-time** *num* | **key-id** *num* [0 | 1] **key** *string*
- **ip ospf passive**
- **ip ospf priority** *value*
- **ip ospf retransmit-interval** *value*
- **ip ospf transmit-delay** *value*

For a complete description of these parameters, see the summary of OSPF port parameters in the next section.

OSPF interface parameters

The following parameters apply to OSPF interfaces.

Area: Assigns an interface to a specific area. You can assign either an IP address or number to represent an OSPF Area ID. If you assign a number, it can be any value from 0 through 2,147,483,647.

Auth-change-wait-time*:* OSPF gracefully implements authentication changes to allow all routers to implement the change and thus prevent disruption to neighbor adjacencies. During the authentication-change interval, both the old and new authentication information is supported. The default authentication-change interval is 300 seconds (5 minutes). You change the interval to a value from 0 through 14400 seconds.

Authentication-key*:* OSPF supports three methods of authentication for each interface—none, simple password, and MD5. Only one method of authentication can be active on an interface at a time. The default authentication value is none, meaning no authentication is performed.

The simple password method of authentication requires you to configure an alphanumeric password on an interface. The simple password setting takes effect immediately. All OSPF packets transmitted on the interface contain this password. Any OSPF packet received on the interface is checked for this password. If the password is not present, then the packet is dropped. The password can be up to eight characters long.

The MD5 method of authentication requires you to configure a key ID and an MD5 Key. The key ID is a number from 1 through 255 and identifies the MD5 key that is being used. The MD5 key can be up to sixteen alphanumeric characters long.

Cost*:* Indicates the overhead required to send a packet across an interface. You can modify the cost to differentiate between 100 Mbps and 1000 Mbps (1 Gbps) links. The default cost is calculated by dividing 100 million by the bandwidth. For 10 Mbps links, the cost is 10. The cost for both 100 Mbps and 1000 Mbps links is 1, because the speed of 1000 Mbps was not in use at the time the OSPF cost formula was devised.

Dead-interval*:* Indicates the number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router down. The value can be from 1 through 65535 seconds. The default is 40 seconds.

Hello-interval: Represents the length of time between the transmission of hello packets. The value can be from 1 through 65535 seconds. The default is 10 seconds.

MD5-authentication activation wait time*:* The number of seconds the Layer 3 Switch waits until placing a new MD5 key into effect. The wait time provides a way to gracefully transition from one MD5 key to another without disturbing the network. The wait time can be from 0 through 14400 seconds. The default is 300 seconds (5 minutes).

MD5-authentication key ID and key*:* A method of authentication that requires you to configure a key ID and an MD5 key. The key ID is a number from 1 through 255 and identifies the MD5 key that is being used. The MD5 key consists of up to 16 alphanumeric characters. The MD5 is encrypted and included in each OSPF packet transmitted.

Passive*:* When you configure an OSPF interface to be passive, that interface does not send or receive OSPF route updates. By default, all OSPF interfaces are active and thus can send and receive OSPF route information. Since a passive interface does not send or receive route information, the interface is in effect a stub network. OSPF interfaces are active by default.

NOTE
This option affects all IP subnets configured on the interface. If you want to disable OSPF updates only on some of the IP subnets on the interface, use the **ospf-ignore** or **ospf-passive** parameter with the **ip address** command. Refer to “Assigning an IP address to an Ethernet port” on page 20.

Priority*:* Allows you to modify the priority of an OSPF router. The priority is used when selecting the designated router (DR) and backup designated routers (BDRs). The value can be from 0 through 255. The default is 1. If you set the priority to 0, the Layer 3 Switch does not participate in DR and BDR election.

Retransmit-interval*:* The time between retransmissions of link-state advertisements (LSAs) to adjacent routers for this interface. The value can be from 0 through 3600 seconds. The default is 5 seconds.

Transit-delay*:* The time it takes to transmit Link State Update packets on this interface. The value can be from 0 through 3600 seconds. The default is 1 second.

Encrypted display of the authentication string or MD5 authentication key

The optional **0 | 1** parameter with the **authentication-key** and **md5-authentication key-id** parameters affects encryption.

For added security, devices encrypt display of the password or authentication string. Encryption is enabled by default. The software also provides an optional parameter to disable encryption of a password or authentication string, on an individual OSPF area or OSPF interface basis.

When encryption of the passwords or authentication strings is enabled, they are encrypted in the CLI regardless of the access level you are using.

The encryption option can be omitted (the default) or can be one of the following:

- **0** – Disables encryption for the password or authentication string you specify with the command. The password or string is shown as clear text in the running-config and the startup-config file. Use this option of you do not want display of the password or string to be encrypted.
- **1** – Assumes that the password or authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE
If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter **0** or **1**. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option **1**, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option **1** followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

If you want to display the authentication string in the output of the **show ip ospf interfac**e command, enter the following commands.

```
Brocade(config)# enable password-display
Brocade# show ip ospf interface 10.1.1.1
```

The **enable password-display** command enables display of the authentication string, but only in the output of the **show ip ospf interface** command. Display of the string is still encrypted in the startup-config file and running-config. Enter the command at the global CONFIG level of the CLI.

Changing the timer for OSPF authentication changes

When you make an OSPF authentication change, the software uses the authentication-change timer to gracefully implement the change. The software implements the change in the following ways:

- Outgoing OSPF packets – After you make the change, the software continues to use the old authentication to send packets, during the remainder of the current authentication-change interval. After this, the software uses the new authentication for sending packets.
- Inbound OSPF packets – The software accepts packets containing the new authentication and continues to accept packets containing the older authentication for two authentication-change intervals. After the second interval ends, the software accepts packets only if they contain the new authentication key.

The default authentication-change interval is 300 seconds (5 minutes). You change the interval to a value from 0 through 14400 seconds.

OSPF provides graceful authentication change for all the following types of authentication changes in OSPF:

- Changing authentication methods from one of the following to another of the following:
  - Simple text password
  - MD5 authentication
  - No authentication
- Configuring a new simple text password or MD5 authentication key
- Changing an existing simple text password or MD5 authentication key

To change the authentication-change interval, enter a command such as the following at the interface configuration level of the CLI.

```
Brocade(config-if-1/1/5)#ip ospf auth-change-wait-time 400
```

Syntax: [**no**] **ip ospf auth-change-wait-time** *secs*

The *secs* parameter specifies the interval and can be from 0 through 14400 seconds. The default is 300 seconds (5 minutes).

NOTE
For backward compatibility, the **ip ospf md5-authentication key-activation-wait-time** *seconds* command is still supported.

Block flooding of outbound LSAs on specific OSPF interfaces

By default, the Layer 3 Switch floods all outbound LSAs on all the OSPF interfaces within an area. You can configure a filter to block outbound LSAs on an OSPF interface. This feature is particularly useful when you want to block LSAs from some, but not all, of the interfaces attached to the area.

After you apply filters to block the outbound LSAs, the filtering occurs during the database synchronization and flooding.

If you remove the filters, the blocked LSAs are automatically re-flooded. You do not need to reset OSPF to re-flood the LSAs.

NOTE
You cannot block LSAs on virtual links.

To apply a filter to an OSPF interface to block flooding of outbound LSAs on the interface, enter the following commands at the Interface configuration level for that interface.

```
Brocade(config-if-1/1/5)#ip ospf database-filter all out
Brocade(config-if-1/1/5)#clear ip ospf all
```

The first command in this example blocks all outbound LSAs on the OSPF interface configured on port 1/1/5. The second command resets OSPF and places the command into effect immediately.

Syntax: [no] ip ospf database-filter all out

To remove the filter, enter a command such as the following.

```
Brocade(config-if-1/1/5)#no ip ospf database-filter all out
```

Configuring an OSPF non-broadcast interface

Layer 3 switches support Non-Broadcast Multi-Access (NBMA) networks. This feature enables you to configure an interface on a Brocade device to send OSPF traffic to its neighbor as unicast packets rather than broadcast packets.

OSPF routers generally use broadcast packets to establish neighbor relationships and broadcast route updates on Ethernet and virtual interfaces (VEs). In this release, as an alternative, you can configure the Brocade device to use unicast packets for this purpose. This can be useful in situations where multicast traffic is not feasible (for example when a firewall does not allow multicast packets).

On a non-broadcast interface, the routers at the other end of this interface must also be configured as non-broadcast and neighbor routers. There is no restriction on the number of routers sharing a non-broadcast interface (for example, through a hub or switch).

NOTE
Only Ethernet interfaces or VEs can be configured as non-broadcast interfaces.

To configure an OSPF interface as a non-broadcast interface, enable the feature on a physical interface or a VE, following the **ip ospf area** statement, and then specify the IP address of the neighbor in the OSPF configuration. The non-broadcast interface configuration must be done on the OSPF routers on both ends of the link.

For example, the following commands configure VE 20 as a non-broadcast interface.

```
Brocade(config)#int ve 20
Brocade(config-vif-20)#ip ospf area 0
Brocade(config-vif-20)#ip ospf network non-broadcast
Brocade(config-vif-20)#exit
```

Syntax: [no] ip ospf network non-broadcast

The following commands specify 10.1.20.1 as an OSPF neighbor address. The address specified must be in the same subnet as a non-broadcast interface.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#neighbor 10.1.20.1
```

For example, to configure the feature in a network with three routers connected by a hub or switch, each router must have the linking interface configured as a non-broadcast interface, and both of the other routers must be specified as neighbors.

The output of the **show ip ospf interface** command has been enhanced to display information about non-broadcast interfaces and neighbors that are configured in the same subnet.

Example of specifying OSPF neighbor address

```
Brocade#show ip ospf interface
v20,OSPF enabled
     IP Address 10.1.20.4, Area 0
     OSPF state BD, Pri 1, Cost 1, Options 2, Type non-broadcast Events 6
     Timers(sec): Transit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 10.1.13.1          Interface Address 10.1.20.5
     BDR: Router ID 10.2.2.1           Interface Address 10.1.20.4
     Neighbor Count = 1, Adjacent Neighbor Count= 2
     Non-broadcast neighbor config: 10.1.20.1, 10.1.20.2, 10.1.20.3, 10.1.20.5,
     Neighbor:         10.1.20.5
     Authentication-Key:None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

In the Type field, “non-broadcast” indicates that this is a non-broadcast interface. When the interface type is non-broadcast, the Non-broadcast neighbor config field displays the neighbors that are configured in the same subnet. If no neighbors are configured in the same subnet, a message such as the following is displayed.

```
***Warning! no non-broadcast neighbor config in 10.1.100.1 255.255.255.0
```

Assigning virtual links

All ABRs (area border routers) must have either a direct or indirect link to the OSPF backbone area (0.0.0.0 or 0). If an ABR does not have a physical link to the area backbone, the ABR can configure a ***virtual link*** to another router within the same area, which has a physical connection to the area backbone.

The path for a virtual link is through an area shared by the neighbor ABR (router with a physical backbone connection), and the ABR requiring a logical connection to the backbone.

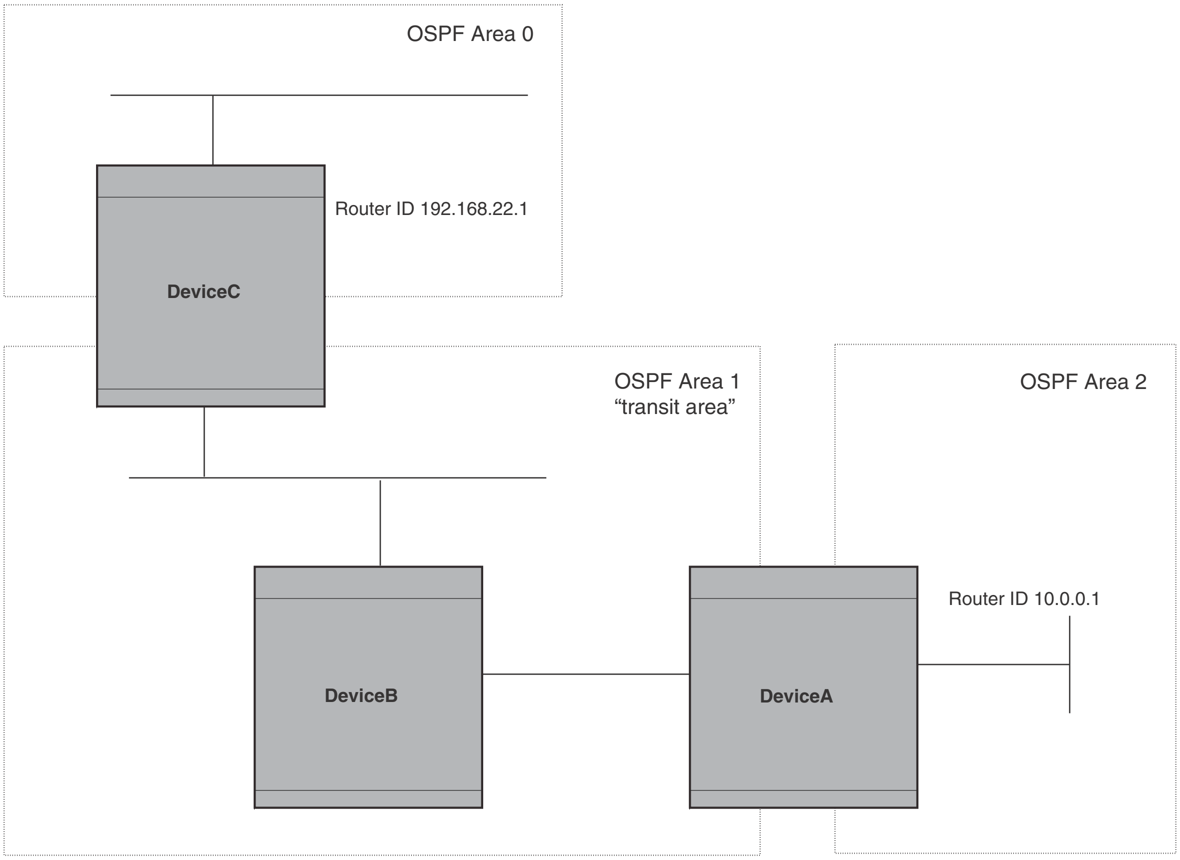
Two parameters fields must be defined for all virtual links—transit area ID and neighbor router:

- The *transit area ID* represents the shared area of the two ABRs and serves as the connection point between the two routers. This number should match the area ID value.
- The *neighbor router* field is the router ID (IP address) of the router that is physically connected to the backbone, when assigned from the router interface requiring a logical connection. When assigning the parameters from the router with the physical connection, the router ID is the IP address of the router requiring a logical connection to the backbone.

NOTE
By default, the Brocade router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, refer to “Changing the router ID” on page 31.

NOTE
When you establish an area virtual link, you must configure it on both of the routers (both ends of the virtual link).

FIGURE 22 Defining OSPF virtual links within a network



Example

Figure 22 shows an OSPF area border router, DeviceA, that is cut off from the backbone area (area 0). To provide backbone access to DeviceA, you can add a virtual link between DeviceA and DeviceC using area 1 as a transit area. To configure the virtual link, you define the link on the router that is at each end of the link. No configuration for the virtual link is required on the routers in the transit area.

To define the virtual link on DeviceA enter the following commands.

```
BrocadeA(config-ospf-router)#area 1 virtual-link 192.168.22.1
BrocadeA(config-ospf-router)#write memory
```

Enter the following commands to configure the virtual link on DeviceC.

```
BrocadeC(config-ospf-router)#area 1 virtual-link 10.0.0.1
BrocadeC(config-ospf-router)#write memory
```

Syntax: **area** *ip-addr* | *num* **virtual-link** *router-id*
[**authentication-key** | **dead-interval** | **hello-interval** | **retransmit-interval** | **transmit-delay** *value*]

The **area** *ip-addr* | *num* parameter specifies the transit area.

The *router-id* parameter specifies the router ID of the OSPF router at the remote end of the virtual link. To display the router ID on a Brocade Layer 3 Switch, enter the **show ip** command.

Refer to “Modifying virtual link parameters” on page 191 for descriptions of the optional parameters.

Modifying virtual link parameters

OSPF has some parameters that you can modify for virtual links. Notice that these are the same parameters as the ones you can modify for physical interfaces.

You can modify default values for virtual links using the following CLI command at the OSPF router level of the CLI, as shown in the following syntax.

Syntax: **area** *num* | *ip-addr* **virtual-link** *ip-addr* [**authentication-key** [0 | 1] *string*] [**dead-interval** *num*]
[**hello-interval** *num*] [**md5-authentication key-activation-wait-time** *num* | **key-id** *num* [0 | 1] **key** *string*]
[**retransmit-interval** *num*] [**transmit-delay** *num*]

The parameters are described in the next section.

Virtual link parameter descriptions

You can modify the following virtual link interface parameters.

Authentication Key: This parameter allows you to assign different authentication methods on a port-by-port basis. OSPF supports three methods of authentication for each interface—none, simple password, and MD5. Only one method of authentication can be active on an interface at a time.

The simple password method of authentication requires you to configure an alphanumeric password on an interface. The password can be up to eight characters long. The simple password setting takes effect immediately. All OSPF packets transmitted on the interface contain this password. All OSPF packets received on the interface are checked for this password. If the password is not present, the packet is dropped.

The MD5 method of authentication encrypts the authentication key you define. The authentication is included in each OSPF packet transmitted.

MD5 Authentication Key: When simple authentication is enabled, the key is an alphanumeric password of up to eight characters. When MD5 is enabled, the key is an alphanumeric password of up to 16 characters that is later encrypted and included in each OSPF packet transmitted. You must enter a password in this field when the system is configured to operate with either simple or MD5 authentication.

MD5 Authentication Key ID: The Key ID is a number from 1 through 255 and identifies the MD5 key that is being used. This parameter is required to differentiate among multiple keys defined on a router.

MD5 Authentication Wait Time: This parameter determines when a newly configured MD5 authentication key is valid. This parameter provides a graceful transition from one MD5 key to another without disturbing the network. All new packets transmitted after the key activation wait time interval use the newly configured MD5 Key. OSPF packets that contain the old MD5 key are accepted for up to five minutes after the new MD5 key is in operation.

The range for the key activation wait time is from 0 through 14400 seconds. The default value is 300 seconds.

Hello Interval: The length of time between the transmission of hello packets. The range is 1 through 65535 seconds. The default is 10 seconds.

Retransmit Interval: The interval between the re-transmission of link state advertisements to router adjacencies for this interface. The range is 0 through 3600 seconds. The default is 5 seconds.

Transmit Delay: The period of time it takes to transmit Link State Update packets on the interface. The range is 0 through 3600 seconds. The default is 1 second.

Dead Interval: The number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router down. The range is 1 through 65535 seconds. The default is 40 seconds.

Encrypted display of the authentication string or MD5 authentication key

The optional **0 | 1** parameter with the **authentication-key** and **md5-authentication key-id** parameters affects encryption.

For added security, FastIron devices encrypt display of the password or authentication string. Encryption is enabled by default. The software also provides an optional parameter to disable encryption of a password or authentication string, on an individual OSPF area or OSPF interface basis.

When encryption of the passwords or authentication strings is enabled, they are encrypted in the CLI regardless of the access level you are using.

The encryption option can be omitted (the default) or can be one of the following:

- **0** – Disables encryption for the password or authentication string you specify with the command. The password or string is shown as clear text in the running-config and the startup-config file. Use this option of you do not want display of the password or string to be encrypted.
- **1** – Assumes that the password or authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE
If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter **0** or **1**. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option **1**, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option 1 followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

Changing the reference bandwidth for the cost on OSPF interfaces

Each interface on which OSPF is enabled has a cost associated with it. The Layer 3 Switch advertises its interfaces and their costs to OSPF neighbors. For example, if an interface has an OSPF cost of ten, the Layer 3 Switch advertises the interface with a cost of ten to other OSPF routers.

By default, an interface OSPF cost is based on the port speed of the interface. The cost is calculated by dividing the reference bandwidth by the port speed. The default reference bandwidth is 100 Mbps, which results in the following default costs:

- 10 Mbps port – 10
- All other port speeds – 1

You can change the reference bandwidth, to change the costs calculated by the software.

The software uses the following formula to calculate the cost.

Cost = reference-bandwidth/interface-speed

If the resulting cost is less than 1, the software rounds the cost up to 1. The default reference bandwidth results in the following costs:

- 10 Mbps port cost = 100/10 = 10
- 100 Mbps port cost = 100/100 = 1
- 1000 Mbps port cost = 100/1000 = 0.10, which is rounded up to 1
- 155 Mbps port cost = 100/155 = 0.65, which is rounded up to 1
- 622 Mbps port cost = 100/622 = 0.16, which is rounded up to 1
- 2488 Mbps port cost = 100/2488 = 0.04, which is rounded up to 1

For 10 Gbps OSPF interfaces, in order to differentiate the costs between 100 Mbps, 1000 Mbps, and 10,000 Mbps interfaces, you can set the auto-cost reference bandwidth to 10000, whereby each slower link is given a higher cost, as follows:

- 10 Mbps port cost = 10000/10 = 1000
- 100 Mbps port cost = 10000/100 = 100
- 1000 Mbps port cost = 10000/1000 = 10
- 10000 Mbps port cost = 10000/10000 = 1

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- Trunk group – The combined bandwidth of all the ports.
- Virtual interface – The combined bandwidth of all the ports in the port-based VLAN that contains the virtual interface.

The default reference bandwidth is 100 Mbps. You can change the reference bandwidth to a value from 1 through 4294967.

If a change to the reference bandwidth results in a cost change to an interface, the Layer 3 Switch sends a link-state update to update the costs of interfaces advertised by the Layer 3 Switch.

NOTE
If you specify the cost for an individual interface, the cost you specify overrides the cost calculated by the software.

Interface types to which the reference bandwidth does not apply

Some interface types are not affected by the reference bandwidth and always have the same cost regardless of the reference bandwidth in use:

- The cost of a loopback interface is always 0.

- The cost of a virtual link is calculated using the Shortest Path First (SPF) algorithm and is not affected by the auto-cost feature.
- The bandwidth for tunnel interfaces is 9 Kbps and is not affected by the auto-cost feature.

Changing the reference bandwidth

To change the reference bandwidth, enter the **auto-cost reference-bandwidth** command at the OSPF configuration level of the CLI.

```
Brocade(config-ospf-router)#auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port cost = 500/10 = 50
- 100 Mbps port cost = 500/100 = 5
- 1000 Mbps port cost = 500/1000 = 0.5, which is rounded up to 1
- 155 Mbps port cost = 500/155 = 3.23, which is rounded up to 4
- 622 Mbps port cost = 500/622 = 0.80, which is rounded up to 1
- 2488 Mbps port cost = 500/2488 = 0.20, which is rounded up to 1

The costs for 10 Mbps, 100 Mbps, and 155 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

Syntax: [**no**] **auto-cost reference-bandwidth** *num*

The *num* parameter specifies the reference bandwidth and can be a value from 1 through 4294967. The default is 100. For 10 Gbps OSPF interfaces, in order to differentiate the costs between 100 Mbps, 1000 Mbps, and 10,000 Mbps interfaces, set the auto-cost reference bandwidth to 10000, whereby each slower link is given a higher cost

To restore the reference bandwidth to its default value and thus restore the default costs of interfaces to their default values, enter the following command.

```
Brocade(config-ospf-router)#no auto-cost reference-bandwidth
```

Defining redistribution filters

Route redistribution imports and translates different protocol routes into a specified protocol type. On Brocade routers, redistribution is supported for static routes, OSPF, RIP, and BGP4. When you configure redistribution for RIP, you can specify that static, OSPF, or BGP4 routes are imported into RIP routes. Likewise, OSPF redistribution supports the import of static, RIP, and BGP4 routes into OSPF routes. BGP4 supports redistribution of static, RIP, and OSPF routes into BGP4.

NOTE
The Layer 3 Switch advertises the default route into OSPF even if redistribution is not enabled, and even if the default route is learned through an IBGP neighbor. IBGP routes (including the default route) are not redistributed into OSPF by OSPF redistribution (for example, by the OSPF **redistribute** command).

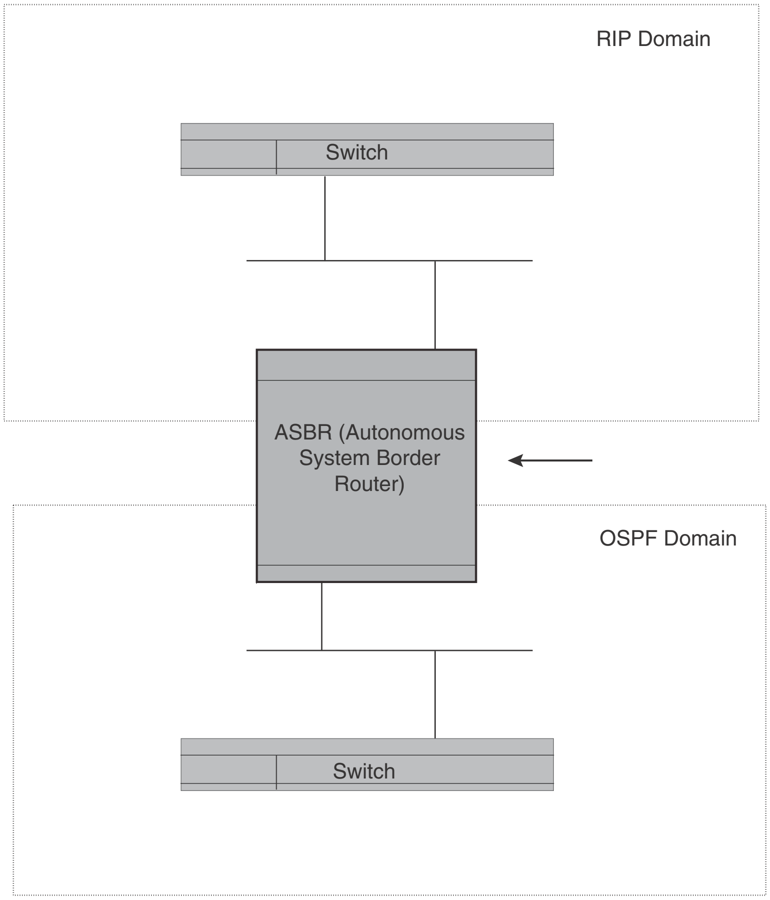
In Figure 23 on page 195, an administrator wants to configure the Layer 3 Switch acting as the ASBR (Autonomous System Boundary Router) between the RIP domain and the OSPF domain to redistribute routes between the two domains.

NOTE
The ASBR must be running both RIP and OSPF protocols to support this activity.

To configure for redistribution, define the redistribution tables with deny and permit redistribution filters. Use the **deny redistribute** and **permit redistribute** commands for OSPF at the OSPF router level.

NOTE
Do not enable redistribution until you have configured the redistribution filters. If you enable redistribution before you configure the redistribution filters, the filters will not take affect and all routes will be distributed.

FIGURE 23 **Redistributing OSPF and static routes to RIP routes**



Example of redefining distribution filters

To configure theLayer 3 Switch acting as an ASBR in Figure 23 to redistribute OSPF, BGP4, and static routes into RIP, enter the following commands.

```
BrocadeASBR(config)#router rip
BrocadeASBR(config-rip-router)#permit redistribute 1 all
BrocadeASBR(config-rip-router)#write memory
```

NOTE
Redistribution is permitted for all routes by default, so the **permit redistribute 1 all** command in the example above is shown for clarity but is not required.

You also have the option of specifying import of just OSPF, BGP4, or static routes, as well as specifying that only routes for a specific network or with a specific cost (metric) be imported, as shown in the following command syntax.

Syntax: **deny** | **permit redistribute** *filter-num* **all** | **bgp** | **connected** | **rip** | **static** [**address** *ip-addr ip-mask* [**match-metric** *value* [**set-metric** *value*]]]

Example

To redistribute RIP, static, and BGP4 routes into OSPF, enter the following commands on the Layer 3 Switch acting as an ASBR.

```
BrocadeASBR(config)#router ospf
BrocadeASBR(config-ospf-router)#permit redistribute 1 all
BrocadeASBR(config-ospf-router)#write memory
```

Syntax: **deny** | **permit redistribute** *filter-num* **all** | **bgp** | **connected** | **rip** | **static**
address *ip-addr ip-mask*
[**match-metric** *value* | **set-metric** *value*]

NOTE
Redistribution is permitted for all routes by default, so the **permit redistribute 1 all** command in the example above is shown for clarity but is not required.

You also have the option of specifying import of just OSPF, BGP4, or static routes, as well as specifying that only routes for a specific network or with a specific cost (metric) be imported, as shown in the following command syntax.

For example, to enable redistribution of RIP and static IP routes into OSPF, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#redistribution rip
Brocade(config-ospf-router)#redistribution static
Brocade(config-ospf-router)#write memory
```

Syntax: [**no**] **redistribution bgp** | **connected** | **rip** | **static** [**route-map** *map-name*]

NOTE
The **redistribution** command does not perform the same function as the **permit redistribute** and **deny redistribute** commands. The **redistribute** commands allow you to control redistribution of routes by filtering on the IP address and network mask of a route. The **redistribution** commands enable redistribution for routes of specific types (static, directly connected, and so on). Configure all your redistribution filters before enabling redistribution.

NOTE
Do not enable redistribution until you have configured the redistribution filters. If you enable redistribution before you configure the redistribution filters, the filters will not take affect and all routes will be distributed.

Preventing specific OSPF routes from being installed in the IP route table

By default, all OSPF routes in the OSPF route table are eligible for installation in the IP route table. You can configure a distribution list to explicitly deny specific routes from being eligible for installation in the IP route table.

NOTE
This feature does not block receipt of LSAs for the denied routes. The Layer 3 Switch still receives the routes and installs them in the OSPF database. The feature only prevents the software from installing the denied OSPF routes into the IP route table.

To configure an OSPF distribution list:

- Configure a standard or extended ACL that identifies the routes you want to deny. Using a standard ACL lets you deny routes based on the destination network, but does not filter based on the network mask. To also filter based on the destination network network mask, use an extended ACL.
- Configure an OSPF distribution list that uses the ACL as input.

NOTE
If you change the ACL after you configure the OSPF distribution list, you must clear the IP route table to place the changed ACL into effect. To clear the IP route table, enter the **clear ip route** command at the Privileged EXEC level of the CLI.

The following sections show how to use the CLI to configure an OSPF distribution list. Separate examples are provided for standard and extended ACLs.

NOTE
The examples show named ACLs. However, you also can use a numbered ACL as input to the OSPF distribution list.

Using a standard ACL as input to the distribution list

To use a standard ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following.

```
Brocade(config)#ip access-list standard no_ip
Brocade(config-std-nACL)#deny 10.0.0.0 0.255.255.255
Brocade(config-std-nACL)#permit any any
Brocade(config-std-nACL)#exit
Brocade(config)#router ospf
Brocade(config-ospf-router)#distribute-list no_ip in
```

The first three commands configure a standard ACL that denies routes to any 10.x.x.x destination network and allows all other routes for eligibility to be installed in the IP route table. The last three commands change the CLI to the OSPF configuration level and configure an OSPF distribution list that uses the ACL as input. The distribution list prevents routes to any 10.x.x.x destination network from entering the IP route table. The distribution list does not prevent the routes from entering the OSPF database.

Syntax: [**no**] **distribute-list** *ACL-name* | *ACL-id* **in** [*interface type*] [*interface number*]

Syntax: [**no**] **ip access-list standard** *ACL-name* | *ACL-id*

Syntax: deny | **permit** *source-ip wildcard*

The *ACL-name* | *ACL-id* parameter specifies the ACL name or ID.

The **in** command applies the ACL to incoming route updates.

The *interface number* parameter specifies the interface number on which to apply the ACL. Enter only one valid interface number. If necessary, use the **show interface brief** command to display a list of valid interfaces. If you do not specify an interface, the Brocade device applies the ACL to all incoming route updates.

If you do not specify an interface type and interface number, the device applies the OSPF distribution list to all incoming route updates.

The **deny** | **permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The *source-ip* parameter specifies the source address for the policy. Because this ACL is input to an OSPF distribution list, the *source-ip* parameter actually is specifying the destination network of the route.

The *wildcard* parameter specifies the portion of the source address to match against. The *wildcard* is in dotted-decimal notation (IP address format). It is a four-part value, where each part is 8 bits (one byte) separated by dots, and each bit is a one or a zero. Each part is a number ranging from 0 to 255, for example 0.0.0.255. Zeros in the mask mean the packet source address must match the *source-ip*. Ones mean any value matches. For example, the *source-ip* and *wildcard* values 10.0.0.0 0.255.255.255 mean that all 4.x.x.x networks match the ACL.

If you want the policy to match on all destination networks, enter **any any**.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “10.0.0.0 0.255.255.255” as “10.0.0.0/8”. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros.

NOTE
If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in “/*mask-bits*” format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** command.

Using an extended ACL as input to the distribution list

You can use an extended ACL with an OSPF distribution list to filter OSPF routes based on the network mask of the destination network.

To use an extended ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following.

```
Brocade(config)#ip access-list extended no_ip
Brocade(config-ext-nACL)#deny ip 10.0.0.0 0.255.255.255 255.255.0.0 0.0.255.255
Brocade(config-ext-nACL)#permit ip any any
Brocade(config-ext-nACL)#exit
Brocade(config)#router ospf
Brocade(config-ospf-router)#distribute-list no_ip in
```

The first three commands configure an extended ACL that denies routes to any 10.x.x.x destination network with a 255.255.0.0 network mask and allows all other routes for eligibility to be installed in the IP route table. The last three commands change the CLI to the OSPF configuration level and configure an OSPF distribution list that uses the ACL as input. The distribution list prevents routes to any 10.x.x.x destination network with network mask 255.255.0.0 from entering the IP route table. The distribution list does not prevent the routes from entering the OSPF database.

Syntax: [**no**] **ip access-list extended** *ACL-name* | *ACL-id*

Syntax: **deny** | **permit** *ip-protocol source-ip wildcard destination-ip wildcard*

The *ACL-name* | *ACL-id* parameter specifies the ACL name or ID.

The **deny** | **permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The *ip-protocol* parameter indicates the type of IP packet you are filtering. When using an extended ACL as input for an OSPF distribution list, specify **ip**.

Because this ACL is input to an OSPF distribution list, the *source-ip* parameter actually specifies the destination network of the route.

The *wildcard* parameter specifies the portion of the source address to match against. The *wildcard* is in dotted-decimal notation (IP address format). It is a four-part value, where each part is 8 bits (one byte) separated by dots, and each bit is a one or a zero. Each part is a number ranging from 0 to 255, for example 0.0.0.255. Zeros in the mask mean the packet source address must match the *source-ip*. Ones mean any value matches. For example, the *source-ip* and *wildcard* values 10.0.0.0 0.255.255.255 mean that all 10.x.x.x networks match the ACL.

If you want the policy to match on all network addresses, enter **any any**.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “10.0.0.0 0.255.255.255” as “10.0.0.0/8”. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros.

NOTE
If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in "/*mask-bits*" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** commands.

Because this ACL is input to an OSPF distribution list, the *destination-ip* parameter actually specifies the subnet mask of the route.

The *wildcard* parameter specifies the portion of the subnet mask to match against. For example, the *destination-ip* and *wildcard* values 255.255.255.255 0.0.0.255 mean that subnet mask /24 and longer match the ACL.

If you want the policy to match on all network masks, enter **any any**.

Modifying the default metric for redistribution

The default metric is a global parameter that specifies the cost applied to all OSPF routes by default. The default value is 10. You can assign a cost from 1 through 15.

NOTE
You also can define the cost on individual interfaces. The interface cost overrides the default cost.

To assign a default metric of 4 to all routes imported into OSPF, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#default-metric 4
```

Syntax: default-metric *value*

The *value* can be from 1 through 16,777,215. The default is 10.

Enabling route redistribution

To enable route redistribution, use one of the following methods.

NOTE
Do not enable redistribution until you have configured the redistribution filters. Otherwise, you might accidentally overload the network with routes you did not intend to redistribute.

To enable redistribution of RIP and static IP routes into OSPF, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#redistribution rip
Brocade(config-ospf-router)#redistribution static
Brocade(config-ospf-router)#write memory
```

Example using a route map

To configure a route map and use it for redistribution of routes into OSPF, enter commands such as the following.

```
Brocade(config)#ip route 10.1.0.0 255.255.0.0 192.168.7.30
Brocade(config)#ip route 10.2.0.0 255.255.0.0 192.168.7.30
Brocade(config)#ip route 10.3.0.0 255.255.0.0 192.168.7.30
Brocade(config)#ip route 10.4.0.0 255.255.0.0 192.168.6.30
Brocade(config)#ip route 10.5.0.0 255.255.0.0 192.168.6.30
Brocade(config)#ip route 10.6.0.0 255.255.0.0 192.168.6.30
Brocade(config)#ip route 10.7.0.0 255.255.0.0 192.168.6.30 5
Brocade(config)#route-map abc permit 1
Brocade(config-routemap abc)#match metric 5
Brocade(config-routemap abc)#set metric 8
Brocade(config-routemap abc)#router ospf
Brocade(config-ospf-router)#redistribution static route-map abc
```

The commands in this example configure some static IP routes, then configure a route map and use the route map for redistributing static IP routes into OSPF.

The **ip route** commands configure the static IP routes. The **route-map** command begins configuration of a route map called “abc”. The number indicates the route map entry (called the “instance”) you are configuring. A route map can contain multiple entries. The software compares packets to the route map entries in ascending numerical order and stops the comparison once a match is found.

The **match** command in the route map matches on routes that have 5 for their metric value (cost). The **set** command changes the metric in routes that match the route map to 8.

The **redistribution static** command enables redistribution of static IP routes into OSPF, and uses route map “abc“ to control the routes that are redistributed. In this example, the route map allows a static IP route to be redistributed into OSPF only if the route has a metric of 5, and changes the metric to 8 before placing the route into the OSPF route table.

Syntax: [**no**] **redistribution bgp** | **connected** | **rip** | **static** [**route-map** *map-name*]

The **bgp** | **connected** | **rip** | **static** parameter specifies the route source.

The **route-map** *map-name* parameter specifies the route map name. The following match parameters are valid for OSPF redistribution:

- **match ip address** | **next-hop** *ACL-num*
- **match metric** *num*
- **match tag** *tag-value*

The following set parameters are valid for OSPF redistribution:

- **set ip next hop** *ip-addr*
- **set metric** [**+** | **-**]*num* | **none**
- **set metric-type type-1** | **type-2**
- **set tag** *tag-value*

NOTE
You must configure the route map before you configure a redistribution filter that uses the route map.

NOTE
When you use a route map for route redistribution, the software disregards the permit or deny action of the route map.

NOTE
For an external route that is redistributed into OSPF through a route map, the metric value of the route remains the same unless the metric is set by a **set metric** command inside the route map. The **default-metric** *num* command has no effect on the route. This behavior is different from a route that is redistributed without using a route map. For a route redistributed without using a route map, the metric is set by the **default-metric** *num* command.

The following command shows the result of the redistribution filter. Because only one of the static IP routes configured above matches the route map, only one route is redistributed. Notice that the route metric is 5 before redistribution but is 8 after redistribution.

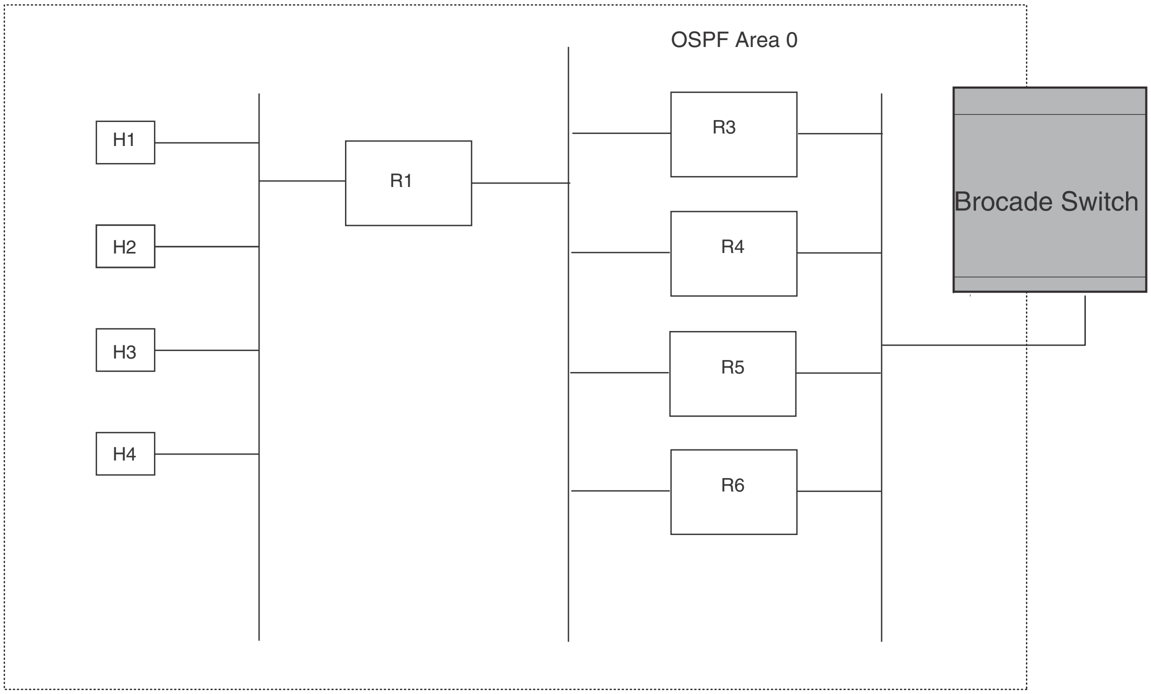
```
Brocade#show ip ospf database external extensive

Index Aging  LS ID           Router          Netmask  Metric    Flag
1     2      10.4.0.0         10.10.10.60     ffff0000 80000008  0000
```

Disabling or re-enabling load sharing

Brocade routers can load share among up to eight equal-cost IP routes to a destination. By default, IP load sharing is enabled. The default is 4 equal-cost paths but you can specify from 2 to 6 paths.

The router software can use the route information it learns through OSPF to determine the paths and costs. Figure 24 shows an example of an OSPF network containing multiple paths to a destination (in this case, R1).

FIGURE 24 Example OSPF network with four equal-cost paths



In the example in Figure 24, the Brocade switch has four paths to R1:

- Brocade Switch->R3
- Brocade Switch->R4
- Brocade Switch->R5
- Brocade Switch->R6

Normally, the Brocade switch will choose the path to the R1 with the lower metric. For example, if R3 metric is 1400 and R4 metric is 600, the Brocade switch will always choose R4.

However, suppose the metric is the same for all four routers in this example. If the costs are the same, the switch now has four equal-cost paths to R1. To allow the switch to load share among the equal cost routes, enable IP load sharing. The software supports four equal-cost OSPF paths by default when you enable load sharing. You can specify from 2 to 6 paths.

NOTE
The Brocade switch is not source routing in these examples. The switch is concerned only with the paths to the next-hop routers, not the entire paths to the destination hosts.

OSPF load sharing is enabled by default when IP load sharing is enabled. To configure IP load sharing parameters, refer to "Configuring IP load sharing" on page 55.

Configuring external route summarization

When the Layer 3 Switch is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to advertise one external route as an aggregate for all redistributed routes that are covered by a specified address range.

When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the Layer 3 Switch, no action is taken if the Layer 3 Switch has already advertised the aggregate route; otherwise the Layer 3 Switch advertises the aggregate route. If an imported route that falls with in a configured address range is removed by the Layer 3 Switch, no action is taken if there are other imported routes that fall with in the same address range; otherwise the aggregate route is flushed.

You can configure up to 32 address ranges. The Layer 3 Switch sets the forwarding address of the aggregate route to zero and sets the tag to zero.

If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually.

If an external LSDB overflow condition occurs, all aggregate routes are flushed out of the AS, along with other external routes. When the Layer 3 Switch exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.

NOTE
If you use redistribution filters in addition to address ranges, the Layer 3 Switch applies the redistribution filters to routes first, then applies them to the address ranges.

NOTE
If you disable redistribution, all the aggregate routes are flushed, along with other imported routes.

To configure a summary address for OSPF routes, enter commands such as the following.

```
Brocade(config-ospf-router)#summary-address 10.1.0.0 255.255.0.0
```

The command in this example configures summary address 10.1.0.0, which includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. For all of these networks, only the address 10.1.0.0 (the parent route) is advertised in external LSAs. However, if the parent route has not been configured with a summary address, or if the summary address for the parent route is configured after the child route, the Layer 3 switch will advertise all routes. For example:

```
router ospf
area 0
summary-address 10.1.1.0 255.255.0.0 -> Advertised
summary-address 10.1.2.0 255.255.0.0 -> Advertised
summary-address 10.0.0.0 255.0.0.0 -> Advertised
```

Syntax: **summary-address** *ip-addr ip-mask*

The *ip-addr* parameter specifies the network address.

The *ip-mask* parameter specifies the network mask.

To display the configured summary addresses, use the **show ip ospf config** command at any level of the CLI. The summary addresses display at the bottom of the output as shown in the following example.

```
Brocade#show ip ospf config

some lines omitted for brevity...

OSPF Redistribution Address Ranges currently defined:
Range-Address    Subnetmask
10.0.0.0          255.0.0.0
10.0.1.0          255.255.255.0
10.0.2.0          255.255.255.0
```

Syntax: **show ip ospf config**

Configuring default route origination

When the Layer 3 Switch is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to automatically generate a default external route into an OSPF routing domain. This feature is called "default route origination" or "default information origination".

By default, Brocade Layer 3 Switches do not advertise the default route into the OSPF domain. If you want the Layer 3 Switch to advertise the OSPF default route, you must explicitly enable default route origination.

When you enable OSPF default route origination, the Layer 3 Switch advertises a type 5 default route that is flooded throughout the AS (except stub areas and NSSAs). In addition, internal NSSA ASBRs advertise their default routes as translatable type 7 default routes.

The Layer 3 Switch advertises the default route into OSPF even if OSPF route redistribution is not enabled, and even if the default route is learned through an IBGP neighbor.

NOTE
Brocade Layer 3 Switches never advertise the OSPF default route, regardless of other configuration parameters, unless you explicitly enable default route origination using the following method.

If the Layer 3 Switch is an ASBR, you can use the "always" option when you enable the default route origination. The always option causes the ASBR to create and advertise a default route if it does not already have one configured.

If default route origination is enabled and you disable it, the default route originated by the Layer 3 Switch is flushed. Default routes generated by other OSPF routers are not affected. If you re-enable the feature, the feature takes effect immediately and thus does not require you to reload the software.

NOTE
The ABR (Layer 3 Switch) will not inject the default route into an NSSA by default and the command described in this section will not cause the Layer 3 Switch to inject the default route into the NSSA. To inject the default route into an NSSA, use the **area** *num* | *ip-addr* **nssa default-information-originate** command. Refer to "Assigning a Not-So-Stubby Area" on page 181.

To enable default route origination, enter the **default-information-originate** command.

```
Brocade(config-ospf-router)#default-information-originate
```

To disable the feature, enter the **no default-information-originate** command.

```
Brocade(config-ospf-router)#no default-information-originate
```

Syntax: [**no**] **default-information-originate** [**always**] [**metric** *value*] [**metric-type** *type*]

The **always** parameter advertises the default route regardless of whether the router has a default route. This option is disabled by default.

The **metric** *value* parameter specifies a metric for the default route. If this option is not used, the default metric is used for the route.

The **metric-type** *type* parameter specifies the external link type associated with the default route advertised into the OSPF routing domain. The *type* can be one of the following:

- 1 – Type 1 external route
- 2 – Type 2 external route

If you do not use this option, the default redistribution metric type is used for the route type.

NOTE
If you specify a metric and metric type, the values you specify are used even if you do not use the **always** option.

Modifying SPF timers

The Layer 3 Switch uses the following timers when calculating the shortest path for OSPF routes:

- SPF delay – When the Layer 3 Switch receives a topology change, the software waits before it starts a Shortest Path First (SPF) calculation. By default, the software waits five seconds. You can configure the SPF delay to a value from 0 through 65535 seconds. If you set the SPF delay to 0 seconds, the software immediately begins the SPF calculation after receiving a topology change.
- SPF hold time – The Layer 3 Switch waits for a specific amount of time between consecutive SPF calculations. By default, the Layer 3 Switch waits ten seconds. You can configure the SPF hold time to a value from 0 through 65535 seconds. If you set the SPF hold time to 0 seconds, the software does not wait between consecutive SPF calculations.

You can set the delay and hold time to lower values to cause the Layer 3 Switch to change to alternate paths more quickly in the event of a route failure. Note that lower values require more CPU processing time.

You can change one or both of the timers. To do so, enter commands such as the following.

```
Brocade(config-ospf-router)#timers spf 10 20
```

The command in this example changes the SPF delay to 10 seconds and changes the SPF hold time to 20 seconds.

Syntax: **timers spf** *delay hold-time*

The *delay* parameter specifies the SPF delay.

The *hold-time* parameter specifies the SPF hold time.

To set the timers back to their default values, enter a command such as the following.

```
Brocade(config-ospf-router)#no timers spf 10 20
```

Modifying the redistribution metric type

The redistribution metric type is used by default for all routes imported into OSPF unless you specify different metrics for individual routes using redistribution filters. Type 2 specifies a big metric (three bytes). Type 1 specifies a small metric (two bytes). The default value is type 2.

To modify the default value to type 1, enter the following command.

```
Brocade(config-ospf-router)#metric-type type1
```

Syntax: **metric-type type1** | **type2**

Administrative distance

Brocade Layer 3 Switches can learn about networks from various protocols, including Border Gateway Protocol version 4 (BGP4), RIP, and OSPF. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned. The default administrative distance for OSPF routes is 110. Refer to “Changing administrative distances” on page 313 for a list of the default distances for all route sources.

The router selects one route over another based on the source of the route information. To do so, the router can use the administrative distances assigned to the sources. You can bias the Layer 3 Switch decision by changing the default administrative distance for RIP routes.

Configuring administrative distance based on route type

You can configure a unique administrative distance for each type of OSPF route. For example, you can use this feature to prefer a static route over an OSPF inter-area route but you also want to prefer OSPF intra-area routes to static routes.

The distance you specify influences the choice of routes when the Layer 3 Switch has multiple routes for the same network from different protocols. The Layer 3 Switch prefers the route with the lower administrative distance.

You can specify unique default administrative distances for the following route types:

- Intra-area routes
- Inter-area routes
- External routes

The default for all these OSPF route types is 110.

NOTE
This feature does not influence the choice of routes within OSPF. For example, an OSPF intra-area route is always preferred over an OSPF inter-area route, even if the intra-area route distance is greater than the inter-area route distance.

To change the default administrative distances for inter-area routes, intra-area routes, and external routes, enter the following command.

```
Brocade(config-ospf-router)#distance external 100
Brocade(config-ospf-router)#distance inter-area 90
Brocade(config-ospf-router)#distance intra-area 80
```

Syntax: [**no**] **distance external** | **inter-area** | **intra-area** *distance*

The **external** | **inter-area** | **intra-area** parameter specifies the route type for which you are changing the default administrative distance.

The *distance* parameter specifies the new distance for the specified route type. Unless you change the distance for one of the route types using commands such as those shown above, the default is 110.

To reset the administrative distance to its system default (110), enter a command such as the following.

```
Brocade(config-ospf-router)#no distance external 100
```

Configuring OSPF group Link State Advertisement pacing

The Layer 3 Switch paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the Layer 3 Switch refreshes and sends out together in one or more packets.

The pacing interval, which is the interval at which the Layer 3 Switch refreshes an accumulated group of LSAs, is configurable to a range from 10 through 1800 seconds (30 minutes). The default is 240 seconds (four minutes). Thus, every four minutes, the Layer 3 Switch refreshes the group of accumulated LSAs and sends the group together in the same packets.

Usage guidelines for configuring OSPF group LSA pacing

The pacing interval is inversely proportional to the number of LSAs the Layer 3 Switch is refreshing and aging. For example, if you have approximately 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might enhance performance slightly.

Changing the LSA pacing interval

To change the LSA pacing interval to two minutes (120 seconds), enter the following command.

```
Brocade(config-ospf-router)#timers lsa-group-pacing 120
```

Syntax: [**no**] **timers lsa-group-pacing** *secs*

The *secs* parameter specifies the number of seconds and can be from 10 through 1800 (30 minutes). The default is 240 seconds (4 minutes).

To restore the pacing interval to its default value, enter the following command.

```
Brocade(config-ospf-router)#no timers lsa-group-pacing
```

Modifying OSPF traps generated

OSPF traps as defined by RFC 1850 are supported on Brocade routers. OSPF trap generation is enabled on the router, by default.

When using the CLI, you can disable all or specific OSPF trap generation by entering the following CLI command.

```
Brocade(config-ospf-router)#no snmp-server trap ospf
```

Syntax: [**no**] **snmp-server trap ospf**

To later re-enable the trap feature, enter **snmp-server trap ospf**.

To disable a specific OSPF trap, enter the command as **no snmp-server trap ospf** *ospf-trap*.

These commands are at the OSPF router level of the CLI.

Here is a summary of OSPF traps supported on Brocade routers, their corresponding CLI commands, and their associated MIB objects from RFC 1850:

- **interface-state-change-trap** – [MIB object: OspfIfstateChange]
- **virtual-interface-state-change-trap** – [MIB object: OspfVirtIfStateChange]
- **neighbor-state-change-trap** – [MIB object:ospfNbrStateChange]
- **virtual-neighbor-state-change-trap** – [MIB object: ospfVirtNbrStateChange]
- **interface-config-error-trap** – [MIB object: ospfIfConfigError]
- **virtual-interface-config-error-trap** – [MIB object: ospfVirtIfConfigError]
- **interface-authentication-failure-trap** – [MIB object: ospfIfAuthFailure]
- **virtual-interface-authentication-failure-trap** – [MIB object: ospfVirtIfAuthFailure]
- **interface-receive-bad-packet-trap** – [MIB object: ospfIfrxBadPacket]
- **virtual-interface-receive-bad-packet-trap** – [MIB object: ospfVirtIfRxBadPacket]
- **interface-retransmit-packet-trap** – [MIB object: ospfTxRetransmit]
- **virtual-interface-retransmit-packet-trap** – [MIB object: ospfVirtIfTxRetransmit]
- **originate-lsa-trap** – [MIB object: ospfOriginateLsa]
- **originate-maxage-lsa-trap** – [MIB object: ospfMaxAgeLsa]
- **link-state-database-overflow-trap** – [MIB object: ospfLsdbOverflow]
- **link-state-database-approaching-overflow-trap** – [MIB object: ospfLsdbApproachingOverflow

Example

To stop an OSPF trap from being collected, use the CLI command: **no trap** *ospf-trap*, at the OSPF router level of the CLI. To disable reporting of the neighbor-state-change-trap, enter the following command.

```
Brocade(config-ospf-router)#no trap neighbor-state-change-trap
```

Example

To reinstate the trap, enter the following command.

```
Brocade(config-ospf-router)#trap neighbor-state-change-trap
```

Syntax: [**no**] **trap** *ospf-trap*

Specifying the types of OSPF Syslog messages to log

You can specify which kinds of OSPF-related Syslog messages are logged. By default, the only OSPF messages that are logged are those indicating possible system errors. If you want other kinds of OSPF messages to be logged, you can configure the Brocade device to log them.

For example, to specify that all OSPF-related Syslog messages be logged, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#log all
```

Syntax: [**no**] **log all** | **adjacency** | **bad_packet** [**checksum**] | **database** | **memory** | **retransmit**

The **all** option causes all OSPF-related Syslog messages to be logged. If you later disable this option with the **no log all** command, the OSPF logging options return to their default settings.

The **adjacency** option logs essential OSPF neighbor state changes, especially on error cases. This option is disabled by default.

The **bad_packet checksum** option logs all OSPF packets that have checksum errors. This option is enabled by default.

The **bad_packet** option logs all other bad OSPF packets. This option is disabled by default.

The **database** option logs OSPF LSA-related information. This option is disabled by default.

The **memory** option logs abnormal OSPF memory usage. This option is enabled by default.

The **retransmit** option logs OSPF retransmission activities. This option is disabled by default.

Modifying the OSPF standard compliance setting

Brocade routers are configured, by default, to be compliant with the RFC 1583 OSPF V2 specification.

To configure a router to operate with the latest OSPF standard, RFC 2178, enter the following commands.

```
Brocade(config)#router ospf
Brocade(config-ospf-router)#no rfc1583-compatibility
```

Syntax: [**no**] **rfc1583-compatibility**

Modifying the exit overflow interval

If a database overflow condition occurs on a router, the router eliminates the condition by removing entries that originated on the router. The exit overflow interval allows you to set how often a Layer 3 Switch checks to see if the overflow condition has been eliminated. The default value is 0. The range is 0 through 86400 seconds (24 hours). If the configured value of the database overflow interval is zero, then the router never leaves the database overflow condition.

NOTE
FastIron devices dynamically allocate OSPF memory as needed. Refer to "Dynamic OSPF memory" on page 176.

To modify the exit overflow interval to 60 seconds, enter the following command.

```
Brocade(config-ospf-router)#database-overflow-interval 60
```

Syntax: **database-overflow-interval** *value*

The *value* can be from 0 through 86400 seconds. The default is 0 seconds.

Configuring an OSPF point-to-point link

In an OSPF point-to-point link, a direct Layer 3 connection exists between a single pair of OSPF routers, without the need for Designated and Backup Designated routers. In a point-to-point link, neighboring routers become adjacent whenever they can communicate directly. In contrast, in broadcast and non-broadcast multi-access (NBMA) networks, the Designated Router and the Backup Designated Router become adjacent to all other routers attached to the network.

Configuration notes and limitations for OSPF point-to-point link

- This feature is supported on 1 Gbps Ethernet and 10 Gbps Ethernet interfaces.
- This feature is supported on physical interfaces. It is not supported on virtual interfaces.
- Brocade supports numbered point-to-point networks, meaning the OSPF router must have an IP interface address which uniquely identifies the router over the network. Brocade does not support unnumbered point-to-point networks.

Configuration syntax for OSFP point-to-point link

To configure an OSPF point-to-point link, enter commands such as the following.

```
Brocade(config)#interface eth 1/1/5
Brocade(config-if-e10000-1/1/5)#ip ospf network point-to-point
```

This command configures an OSPF point-to-point link on Interface 5 in slot 1.

Syntax: [**no**] **ip ospf network point-to-point**

Viewing configured OSPF point-to-point links

Refer to "Displaying OSPF neighbor information" on page 217 and "Displaying OSPF interface information" on page 219.

Configuring OSPF graceful restart

By default, OSPF graceful restart is enabled for the global instance.

For information about how to display OSPF graceful restart information, refer to "Displaying OSPF graceful restart information" on page 226.

Enabling and disabling OSPF graceful restart

OSPF graceful restart is enabled by default on a Layer 3 switch. To disable it, use the following commands.

```
Brocade(config)# router ospf
Brocade(config-ospf-router)# no graceful-restart
```

To re-enable OSPF graceful restart after it has been disabled, enter the following commands.

```
Brocade(config)# router ospf
Brocade(config-ospf-router)# graceful-restart
```

Syntax: [**no**] **graceful-restart**

Configuring the OSPF graceful restart time

Use the following commands to specify the maximum amount of time advertised to a neighbor router to maintain routes from and forward traffic to a restarting router.

```
Brocade(config) router ospf
Brocade(config-ospf-router)# graceful-restart restart-time 120
```

Syntax: [**no**] **graceful-restart restart-time** *seconds*

The *seconds* variable sets the maximum restart wait time advertised to neighbors.

Possible values are from 10 through 1800 seconds.

The default value is 120 seconds.

Disabling OSPF graceful restart helper mode

By default, a Layer 3 switch supports other restarting routers as a helper. You can prevent your router from participating in OSPF graceful restart by using the following commands.

```
Brocade(config) router ospf
Brocade(config-ospf-router)# graceful-restart helper-disable
```

Syntax: [**no**] **graceful-restart helper-disable**

This command disables OSPF graceful restart helper mode.

The default behavior is to help the restarting neighbors.

Clearing OSPF information

The following kinds of OSPF information can be cleared from a Brocade OSPF link state database and OSPF routing table:

- Routes received from OSPF neighbors. You can clear routes from all OSPF neighbors, or an individual OSPF neighbor, specified either by the neighbor IP address or its router ID
- OSPF topology information, including all routes in the OSPF routing table
- All routes in the OSPF routing table that were redistributed from other protocols
- OSPF area information, including routes received from OSPF neighbors within an area, as well as routes imported into the area. You can clear area information for all OSPF areas, or for a specified OSPF area

The OSPF information is cleared dynamically when you enter the command; you do not need to remove statements from the Brocade configuration or reload the software for the change to take effect.

Clearing OSPF neighbor information

To clear information on the Brocade device about all OSPF neighbors, enter the following command.

```
Brocade#clear ip ospf neighbor
```

Syntax: **clear ip ospf neighbor** [**ip** *ip-addr* | **id** *ip-addr*]

This command clears all OSPF neighbors and the OSPF routes exchanged with the neighbors in the Brocade OSPF link state database. After this information is cleared, adjacencies with all neighbors are re-established, and routes with these neighbors exchanged again.

To clear information on the Brocade device about OSPF neighbor 10.10.10.1, enter the following command.

```
Brocade#clear ip ospf neighbor ip 10.10.10.1
```

This command clears the OSPF neighbor and the OSPF routes exchanged with neighbor 10.10.10.1 in the OSPF link state database in the Brocade device. After this information is cleared, the adjacency with the neighbor is re-established, and routes are exchanged again.

The neighbor router can be specified either by its IP address or its router ID. To specify the neighbor router using its IP address, use the **ip** *ip-addr* parameter. To specify the neighbor router using its router ID, use the **id** *ip-addr* parameter.

Clearing OSPF topology information

To clear OSPF topology information on the Brocade device, enter the following command.

```
Brocade#clear ip ospf topology
```

Syntax: **clear ip ospf topology**

This command clears all OSPF routes from the OSPF routing table, including intra-area, (which includes ABR and ASBR intra-area routes), inter-area, external type 1, external type 2, OSPF default, and OSPF summary routes.

After you enter this command, the OSPF routing table is rebuilt, and valid routes are recomputed from the OSPF link state database. When the OSPF routing table is cleared, OSPF routes in the global routing table are also recalculated. If redistribution is enabled, the routes are imported again.

Clearing redistributed routes from the OSPF routing table

To clear all routes in the OSPF routing table that were redistributed from other protocols, enter the following command.

```
Brocade#clear ip ospf redistribution
```

Syntax: **clear ip ospf redistribution**

This command clears all routes in the OSPF routing table that are redistributed from other protocols, including direct connected, static, RIP, and BGP. To import redistributed routes from other protocols, use the **redistribution** command at the OSPF configuration level.

Clearing information for OSPF areas

To clear information on the Brocade device about all OSPF areas, enter the following command.

```
Brocade#clear ip ospf
```

This command clears all OSPF areas, all OSPF neighbors, and the entire OSPF routing table. After this information has been cleared, adjacencies with all neighbors are re-established, and all OSPF routes are re-learned.

To clear information on the Brocade device about OSPF area 1, enter the following command.

```
Brocade#clear ip ospf area 1
```

This command clears information about the specified area ID. Information about other OSPF areas is not affected. The command clears information about all OSPF neighbors belonging to the specified area, as well as all routes imported into the specified area. Adjacencies with neighbors belonging to the area are re-established, and routes imported into the area are re-learned.

Syntax: **clear ip ospf** [**area** *area-id*]

The *area-id* can be specified in decimal format or in IP address format.

Displaying OSPF information

You can use CLI commands to display the following OSPF information:

- Trap, area, and interface information – refer to "Displaying general OSPF configuration information" on page 214.
- CPU utilization statistics – refer to "Displaying CPU utilization statistics" on page 215.
- Area information – refer to "Displaying OSPF area information" on page 216.
- Neighbor information – refer to "Displaying OSPF neighbor information" on page 217.
- Interface information – refer to "Displaying OSPF interface information" on page 219.
- Route information – refer to "Displaying OSPF route information" on page 220.
- External link state information – refer to "Displaying OSPF external link state information" on page 222.
- Link state information – refer to "Displaying OSPF link state information" on page 223.
- Virtual Neighbor information – refer to "Displaying OSPF virtual neighbor information" on page 224.
- Virtual Link information – refer to "Displaying OSPF virtual link information" on page 224.
- ABR and ASBR information – refer to "Displaying OSPF ABR and ASBR information" on page 225.
- Trap state information – refer to "Displaying OSPF trap status" on page 225.
- OSPF graceful restart - refer to "Displaying OSPF graceful restart information" on page 226.

Displaying general OSPF configuration information

To display general OSPF configuration information, enter the following command at any CLI level.

```
Brocade#show ip ospf config
Router OSPF: Enabled
Redistribution: Disabled
Default OSPF Metric: 10
OSPF Redistribution Metric: Type2

OSPF External LSA Limit: 25000

OSPF Database Overflow Interval: 0

RFC 1583 Compatibility: Enabled
```

```
Router id: 192.168.2.1
Interface State Change Trap:                      Enabled
Virtual Interface State Change Trap:              Enabled
Neighbor State Change Trap:                       Enabled
Virtual Neighbor State Change Trap:               Enabled
Interface Configuration Error Trap:               Enabled
Virtual Interface Configuration Error Trap:       Enabled
Interface Authentication Failure Trap:            Enabled
Virtual Interface Authentication Failure Trap:    Enabled
Interface Receive Bad Packet Trap:                Enabled
Virtual Interface Receive Bad Packet Trap:        Enabled
Interface Retransmit Packet Trap:                 Enabled
Virtual Interface Retransmit Packet Trap:         Enabled
Originate LSA Trap:                               Enabled
Originate MaxAge LSA Trap:                        Enabled
Link State Database Overflow Trap:                Enabled
Link State Database Approaching Overflow Trap:    Enabled

OSPF Area currently defined:
Area-ID          Area-Type Cost
0                      normal    0

OSPF Interfaces currently defined:
Ethernet Interface: 1/1/2-1/1/3
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf area 0

Ethernet Interface: v1
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf area 0

Ethernet Interface: 1/2/1
ip ospf auth-change-wait-time 300
ip ospf cost 40
ip ospf area 0
```

Syntax: show ip ospf config

Displaying CPU utilization statistics

You can display CPU utilization statistics for OSPF and other IP protocols.

To display CPU utilization statistics for OSPF for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI.

```
Brocade#show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22             9
BGP              0.04      0.06      0.08      0.14             13
GVRP             0.00      0.00      0.00      0.00             0
ICMP             0.00      0.00      0.00      0.00             0
IP               0.00      0.00      0.00      0.00             0
OSPF             0.03      0.06      0.09      0.12             11
RIP              0.00      0.00      0.00      0.00             0
STP              0.00      0.00      0.00      0.00             0
VRRP             0.00      0.00      0.00      0.00             0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example.

```
Brocade#show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00              0
BGP              0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.00      0.00      0.00              1
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter the **show process cpu** command.

```
Brocade#show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00        0
BGP              0.00        0
GVRP             0.00        0
ICMP             0.01        1
IP               0.00        0
OSPF             0.00        0
RIP              0.00        0
STP              0.01        0
VRRP             0.00        0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: **show process cpu** [*num*]

The *num* parameter specifies the number of seconds and can be from 1 through 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying OSPF area information

To display OSPF area information, enter the **show ip ospf area** command at any CLI level.

```
Brocade#show ip ospf area
Indx  Area      Type  Cost  SPFR ABR ASBR LSA Chksum(Hex)
1   0.0.0.0     normal  0    1     0     0   1   0000781f
2  192.168.60.0 normal  0    1     0     0   1   0000fee6
3  192.168.80.0 stub    1    1     0     0   2   000181cd
```

Syntax: **show ip ospf area** [*area-id*] | [*num*]

The *area-id* parameter shows information for the specified area.

The *num* parameter displays the entry that corresponds to the entry number you enter. The entry number identifies the entry position in the area table.

This display shows the following information.

TABLE 36 CLI display of OSPF area information

Field	Definition
Indx	The row number of the entry in the router OSPF area table.
Area	The area number.
Type	The area type, which can be one of the following: • nssa • normal • stub
Cost	The area cost.
SPFR	The SPFR value.
ABR	The ABR number.
ASBR	The ABSR number.
LSA	The LSA number.
Chksum(Hex)	The checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The Layer 3 Switch uses the checksum to verify that the packet is not corrupted.

Displaying OSPF neighbor information

To display OSPF neighbor information, enter the following command at any CLI level.

```
Brocade#show ip ospf neighbor
Port     Address         Pri State      Neigh Address   Neigh ID
1/1/8    192.168.7.251    1   full       192.168.7.200 192.168.1.220
```

To display detailed OSPF neighbor information, enter the following command at any CLI level.

```
Brocade#show ip ospf neighbor detail

Port        Address         Pri State      Neigh Address   Neigh ID    Ev Op Cnt
1/1/9        10.2.0.2        1  FULL/DR    10.2.0.1      192.168.2.2      6  2  0
    Second-to-dead:39
1/1/10        10.3.0.2        1   FULL/BDR   10.3.0.1       192.168.3.3      5
2  0
    Second-to-dead:36
1/1/1-1/1/8 10.5.0.1        1   FULL/DR    10.5.0.2        192.168.16.16  6  2  0
    Second-to-dead:33
1/2/1-1/2/2    10.2.0.1        1   FULL/DR 10.2.0.2        192.168.15.15  6  2  0
    Second-to-dead:33
```

Syntax: **show ip ospf neighbor** [**router-id** *ip-addr*] | [*num*] | [**detail**]

The **router-id** *ip-addr* parameter displays only the neighbor entries for the specified router.

The *num* parameter displays only the entry in the specified index position in the neighbor table. For example, if you enter “1”, only the first entry in the table is displayed.

The **detail** parameter displays detailed information about the neighbor routers.

These displays show the following information.

TABLE 37 CLI display of OSPF neighbor information

Field	Description
Port	The port through which the Layer 3 Switch is connected to the neighbor. The port on which an OSPF point-to-point link is configured.
Address	The IP address of this Layer 3 Switch interface with the neighbor.
Pri	The OSPF priority of the neighbor: • For multi-access networks, the priority is used during election of the Designated Router (DR) and Backup designated Router (BDR). • For point-to-point links, this field shows one of the following values: • 1 = point-to-point link • 3 = point-to-point link with assigned subnet
State	The state of the conversation between the Layer 3 Switch and the neighbor. This field can have one of the following values: • Down – The initial state of a neighbor conversation. This value indicates that there has been no recent information received from the neighbor. • Attempt – This state is only valid for neighbors attached to non-broadcast networks. It indicates that no recent information has been received from the neighbor. • Init – A Hello packet has recently been seen from the neighbor. However, bidirectional communication has not yet been established with the neighbor. (The router itself did not appear in the neighbor's Hello packet.) All neighbors in this state (or higher) are listed in the Hello packets sent from the associated interface. • 2-Way – Communication between the two routers is bidirectional. This is the most advanced state before beginning adjacency establishment. The Designated Router and Backup Designated Router are selected from the set of neighbors in the 2-Way state or greater. • ExStart – The first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is the master, and to decide upon the initial Database Description (DD) sequence number. Neighbor conversations in this state or greater are called adjacencies. • Exchange – The router is describing its entire link state database by sending Database Description packets to the neighbor. Each Database Description packet has a DD sequence number, and is explicitly acknowledged. Only one Database Description packet can be outstanding at any time. In this state, Link State Request packets can also be sent asking for the neighbor's more recent advertisements. All adjacencies in Exchange state or greater are used by the flooding procedure. In fact, these adjacencies are fully capable of transmitting and receiving all types of OSPF routing protocol packets. • Loading – Link State Request packets are sent to the neighbor asking for the more recent advertisements that have been discovered (but not yet received) in the Exchange state. • Full – The neighboring routers are fully adjacent. These adjacencies will now appear in router links and network link advertisements.
Neigh Address	The IP address of the neighbor: For point-to-point links, the value is as follows: • If the Pri field is "1", this value is the IP address of the neighbor router interface. • If the Pri field is "3", this is the subnet IP address of the neighbor router interface.
Neigh ID	The neighbor router ID.
Ev	The number of times the neighbor state changed.

TABLE 37 CLI display of OSPF neighbor information (Continued)

Field	Description
Opt	The sum of the option bits in the Options field of the Hello packet. This information is used by Brocade technical support. Refer to Section A.2 in RFC 2178 for information about the Options field in Hello packets.
Cnt	The number of LSAs that were retransmitted.
Second-to-dead	The amount of time the Brocade device will wait for a HELLO message from each OSPF neighbor before assuming the neighbor is dead.

Displaying OSPF interface information

To display OSPF interface information, enter the **show ip ospf interface** command at any CLI level.

```
Brocade#show ip ospf interface 192.168.1.1

Ethernet 1/2/1,OSPF enabled
     IP Address 192.168.1.1, Area 0
     OSPF state ptr2ptr, Pri 1, Cost 1, Options 2, Type pt-2-pt Events 1
     Timers(sec): Transit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 0.0.0.0           Interface Address 0.0.0.0
     BDR: Router ID 0.0.0.0           Interface Address 0.0.0.0
     Neighbor Count = 0, Adjacent Neighbor Count= 1
     Neighbor: 10.2.2.2
     Authentication-Key:None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

Syntax: **show ip ospf interface** [*ip-addr*]

The *ip-addr* parameter displays the OSPF interface information for the specified IP address.

The following table defines the highlighted fields shown in the above example output of the **show ip ospf interface** command.

TABLE 38 Output of the show ip ospf interface command

Field	Definition
IP Address	The IP address of the interface.
OSPF state	ptr2ptr (point to point)
Pri	The link ID as defined in the router-LSA. This value can be one of the following: • 1 = point-to-point link • 3 = point-to-point link with an assigned subnet
Cost	The configured output cost for the interface.
Options	OSPF Options (Bit7 - Bit0): • unused:1 • opaque:1 • summary:1 • dont_propagate:1 • nssa:1 • multicast:1 • externals:1 • tos:1

TABLE 38 Output of the **show ip ospf interface** command (Continued)

Field	Definition
Type	The area type, which can be one of the following: • Broadcast = 0x01 • NBMA = 0x02 • Point to Point = 0x03 • Virtual Link = 0x04 • Point to Multipoint = 0x05
Events	OSPF Interface Event: • Interface_Up = 0x00 • Wait_Timer = 0x01 • Backup_Seen = 0x02 • Neighbor_Change = 0x03 • Loop_Indication = 0x04 • Unloop_Indication = 0x05 • Interface_Down = 0x06 • Interface_Passive = 0x07
Adjacent Neighbor Count	The number of adjacent neighbor routers.
Neighbor:	The neighbor router ID.

Displaying OSPF route information

To display OSPF route information for the router, enter the **show ip ospf routes** command at any CLI level.

```
Brocade#show ip ospf routes
Index Destination     Mask            Path_Cost Type2_Cost Path_Type
1     192.168.7.0      255.255.255.0   1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      192.168.1.220    10.95.7.251    Network   Valid      00000000  7000
      Paths Out_Port  Next_Hop        Type      Arp_Index  State
      1     1/1/5     192.168.7.250    OSPF      8          84 00

Index Destination     Mask            Path_Cost Type2_Cost Path_Type
2     10.3.63.0       255.255.255.0   11        0          Inter
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      192.168.7.250    10.3.63.0       Network   Valid      00000000  0000
      Paths Out_Port  Next_Hop        Type      Arp_Index  State
      1     1/1/6     192.168.7.250    OSPF      8          84 00
```

Syntax: **show ip ospf routes** [*ip-addr*]

The *ip-addr* parameter specifies a destination IP address. If you use this parameter, only the route entries for that destination are shown.

This display shows the following information.

TABLE 39 CLI Display of OSPF route information

Field	Definition
Index	The row number of the entry in the router OSPF route table.
Destination	The IP address of the route's destination.

TABLE 39 CLI Display of OSPF route information (Continued)

Field	Definition
Mask	The network mask for the route.
Path_Cost	The cost of this route path. (A route can have multiple paths. Each path represents a different exit port for the Layer 3 Switch.)
Type2_Cost	The type 2 cost of this path.
Path_Type	The type of path, which can be one of the following: • Inter – The path to the destination passes into another area. • Intra – The path to the destination is entirely within the local area. • External1 – The path to the destination is a type 1 external route. • External2 – The path to the destination is a type 2 external route.
Adv_Router	The OSPF router that advertised the route to this Brocade Layer 3 Switch.
Link-State	The link state from which the route was calculated.
Dest_Type	The destination type, which can be one of the following: • ABR – Area Border Router • ASBR – Autonomous System Boundary Router • Network – the network
State	The route state, which can be one of the following: • Changed • Invalid • Valid This information is used by Brocade technical support.
Tag	The external route tag.
Flags	State information for the route entry. This information is used by Brocade technical support.
Paths	The number of paths to the destination.
Out_Port	The router port through which the Layer 3 Switch reaches the next hop for this route path.
Next_Hop	The IP address of the next-hop router for this path.
Type	The route type, which can be one of the following: • OSPF • Static Replaced by OSPF
Arp_Index	The index position in the ARP table of the ARP entry for this path's IP address.
State	State information for the path. This information is used by Brocade technical support.

Displaying the routes that have been redistributed into OSPF

You can display the routes that have been redistributed into OSPF. To display the redistributed routes, enter the **show ip ospf redistribute route** command at any level of the CLI.

```
Brocade#show ip ospf redistribute route
10.3.0.0 255.255.0.0 static
10.1.0.0 255.255.0.0 static
10.11.61.0 255.255.255.0 connected
10.4.0.0 255.255.0.0 static
```

In this example, four routes have been redistributed. Three of the routes were redistributed from static IP routes and one route was redistributed from a directly connected IP route.

Syntax: **show ip ospf redistribute route** [*ip-addr ip-mask*]

The *ip-addr ip-mask* parameter specifies a network prefix and network mask. Here is an example.

```
Brocade#show ip ospf redistribute route 10.1.0.0 255.255.0.0
  10.1.0.0 255.255.0.0 static
```

Displaying OSPF external link state information

To display external link state information, enter the **show ip ospf database external-link-state** command at any CLI level.

```
Brocade#show ip ospf database external-link-state

Index Aging  LS ID            Router          Netmask  Metric    Flag
1     1794   10.168.64.0      192.168.0.3     ffffe000 000003e8  b500 0.0.0.0
2     1794   10.215.0.0       192.168.0.3     ffff0000 000003e8  b500 0.0.0.0
3     1794   10.27.250.0      192.168.0.3     fffffe00 000003e8  b500 0.0.0.0
4     1794   10.24.23.0       192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
5     1794   10.21.52.0       192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
6     1794   10.18.81.0       192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
7     1794   10.15.110.0      192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
8     1794   10.12.139.0      192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
9     1794   10.9.168.0       192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
```

Syntax: **show ip ospf database external-link-state** [**advertise** *num*] | [**extensive**] | [**link-state-id** *ip-addr*] | [**router-id** *ip-addr*] | [**sequence-number** *num(Hex)*] | [**status** *num*]

The **advertise** *num* parameter displays the hexadecimal data in the specified LSA packet. The *num* parameter identifies the LSA packet by its position in the router External LSA table. To determine an LSA packet position in the table, enter the **show ip ospf external-link-state** command to display the table. Refer to "Displaying the data in an LSA" on page 224 for an example.

The **extensive** option displays the LSAs in decrypted format.

NOTE
You cannot use the **extensive** option in combination with other display options. The entire database is displayed.

The **link-state-id** *ip-addr* parameter displays the External LSAs for the LSA source specified by *IP-addr*.

The **router-id** *ip-addr* parameter shows the External LSAs for the specified OSPF router.

The **sequence-number** *num(Hex*) parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

The **status** *num* option shows status information.

This OSPF external link state display shows the following information.

TABLE 40 CLI display of OSPF external link state information

Field	Definition
Area ID	The OSPF area the router is in.
Aging	The age of the LSA, in seconds.

TABLE 40 CLI display of OSPF external link state information (Continued)

Field	Definition
LS ID	The ID of the link-state advertisement from which the Layer 3 Switch learned this route.
Router	The router IP address.
Seq(hex)	The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps it with a sequence number to enable the Layer 3 Switch and other OSPF routers to determine which LSA for a given route is the most recent.
Chksum	A checksum for the LSA packet, which is based on all the fields in the packet except the age field. The Layer 3 Switch uses the checksum to verify that the packet is not corrupted.
Type	The route type, which is always EXTR (external).

Displaying OSPF link state information

To display link state information, enter the **show ip ospf database link-state** command at any CLI level.

```
Brocade#show ip ospf database link-state
```

Syntax: **show ip ospf database link-state** [**advertise** *num*] | [**asbr**] | [**extensive**] | [**link-state-id** *ip-addr*] | [**network**] | [**nssa**] | [**opaque-area**] | [**router**] | [**router-id** *ip-addr*] | [**sequence-number** *num(Hex)*] | [**status** *num*] | [**summary**]

The **advertise** *num* parameter displays the hexadecimal data in the specified LSA packet. The *num* parameter identifies the LSA packet by its position in the router External LSA table. To determine an LSA packet position in the table, enter the **show ip ospf external-link-state** command to display the table. Refer to “Displaying the data in an LSA” on page 224 for an example.

The **asbr** option shows ASBR information.

The **extensive** option displays the LSAs in decrypted format.

NOTE
You cannot use the **extensive** option in combination with other display options. The entire database is displayed.

The **link-state-id** *ip-addr* parameter displays the External LSAs for the LSA source specified by *IP-addr*.

The **network** option shows network information.

The **nssa** option shows network information.

The **opaque-area** option shows information for opaque areas.

The **router-id** *ip-addr* parameter shows the External LSAs for the specified OSPF router.

The **sequence-number** *num(Hex)* parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

The **status** *num* option shows status information.

The **summary** option shows summary information.

Displaying the data in an LSA

You can use the CLI to display the data the Layer 3 Switch received in a specific External LSA packet or other type of LSA packet. For example, to display the LSA data in entry 3 in the External LSA table, enter the following command.

```
Brocade#show ip ospf database external-link-state advertise 3
Index Aging  LS ID            Router          Netmask   Metric    Flag
3     619    10.27.250.0      192.168.0.3      fffffe00 000003e8  b500 0.0.0.0
  LSA Header:  age: 619, options: 0x02, seq-nbr: 0x80000003, length: 36
  NetworkMask: 255.255.254.0
  TOS 0:  metric_type: 1, metric: 1000
          forwarding_address: 0.0.0.0
          external_route_tag: 0
```

Syntax: **show ip ospf database external-link-state** [**advertise** *num*] | [**link-state-id** *ip-addr*] | [**router-id** *ip-addr*] | [**sequence-number** *num(Hex)*] | [**status** *num*]

To determine an external LSA or other type of LSA index number, enter one of the following commands to display the appropriate LSA table:

- **show ip ospf database link-state advertise** *num* – This command displays the data in the packet for the specified LSA.
- **show ip ospf database external-link-state advertise** *num* – This command displays the data in the packet for the specified external LSA.

For example, to determine an external LSA index number, enter the **show ip ospf external-link-state** command.

```
Brocade#show ip ospf external-link-state

Index Aging  LS ID            Router          Netmask  Metric    Flag
1     1809   10.18.81.0       192.168.103.6   ffffff00 000003e8  b500 0.0.0.0
2     8      10.27.250.0      192.168.103.6   fffffe00 000003e8  b500 0.0.0.0
3     8      10.215.0.0       192.168.103.6   ffff0000 000003e8  b500 0.0.0.0
4     18     10.33.192.0      192.168.102.6   fffffc00 000003e8  b500 0.0.0.0
5     959    10.9.168.0       192.168.102.6   ffffff00 00002710  b500 0.0.0.0
6     1807   10.3.226.0       192.168.0.3     ffffff00 000003e8  b500 0.0.0.0
7     1809   10.6.197.0       192.168.3.3     ffffff00 000003e8  b500 0.0.0.0
```

Displaying OSPF virtual neighbor information

To display OSPF virtual neighbor information, enter the **show ip ospf virtual-neighbor** command at any CLI level.

```
Brocade#show ip ospf virtual-neighbor
```

Syntax: **show ip ospf virtual-neighbor** [*num*]

The *num* parameter displays the table beginning at the specified entry number.

Displaying OSPF virtual link information

To display OSPF virtual link information, enter the **show ip ospf virtual-link** command at any CLI level.

```
Brocade#show ip ospf virtual-link
```

Syntax: **show ip ospf virtual-link** [*num*]

The *num* parameter displays the table beginning at the specified entry number.

Displaying OSPF ABR and ASBR information

To display OSPF ABR and ASBR information, enter the **show ip ospf border-routers** command at any CLI level.

```
Brocade#show ip ospf border-routers
```

Syntax: **show ip ospf border-routers** [*ip-addr*]

The *ip-addr* parameter displays the ABR and ASBR entries for the specified IP address.

Displaying OSPF trap status

All traps are enabled by default when you enable OSPF. To disable or re-enable an OSPF trap, refer to “Modifying OSPF traps generated” on page 208.

To display the state of each OSPF trap, enter the **show ip ospf trap** command at any CLI level.

```
Brocade#show ip ospf trap
Interface State Change Trap:                      Enabled
Virtual Interface State Change Trap:              Enabled
Neighbor State Change Trap:                       Enabled
Virtual Neighbor State Change Trap:               Enabled
Interface Configuration Error Trap:               Enabled
Virtual Interface Configuration Error Trap:       Enabled
Interface Authentication Failure Trap:            Enabled
Virtual Interface Authentication Failure Trap:    Enabled
Interface Receive Bad Packet Trap:                Enabled
Virtual Interface Receive Bad Packet Trap:        Enabled
Interface Retransmit Packet Trap:                 Enabled
Virtual Interface Retransmit Packet Trap:         Enabled
Originate LSA Trap:                               Enabled
Originate MaxAge LSA Trap:                        Enabled
Link State Database Overflow Trap:                Enabled
Link State Database Approaching Overflow Trap:    Enabled
```

Syntax: **show ip ospf trap**

Displaying OSPF graceful restart information

To display OSPF graceful restart information for OSPF neighbors, use the **show ip ospf neighbors** command.

```
Brocade#show ip ospf neighbors
Port  Address         Pri   State       Neigh Address   Neigh ID        Ev Opt Cnt
1/1/2 192.168.50.10    0   FULL/OTHER   192.168.50.1    10.10.10.30     21 66  0
  < in graceful restart state,  helping 1, timer 60 sec >
```

Syntax: **show ip ospf neighbor**

Use the following command to display Type 9 grace LSAs on a Brocade Layer 3 switch.

```
Brocade#show ip ospf database grace-link-state
Graceful Link States
Area  Interface   Adv Rtr  Age Seq(Hex) Prd Rsn  Nbr Intf IP
0     eth 1/1/2   10.2.2.2  7   80000001 60  SW   10.1.1.2
```

Syntax: **show ip ospf database grace-link-state**

Table 41 defines the fields in the show output.

TABLE 41 CLI display of OSPF database grace LSA information

Field	Definition
Area	The OSPF area that the interface configured for OSPF graceful restart is in.
Interface	The interface that is configured for OSPF graceful restart.
Adv Rtr	The ID of the advertised route.
Age	The age of the LSA in seconds.
Seq (Hex)	The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps the LSA with a sequence number. This number enables the FastIron and other OSPF routers to determine the most recent LSA for a given route.
Prd	The grace period. The number of seconds that the neighbor routers should continue to advertise the router as fully adjacent, regardless of the state of database synchronization between the router and its neighbors. Since this time period begins when the grace LSA LS age is equal to 0, the grace period terminates when either the LS age of the grace LSA exceeds the value of a grace period or the grace LSA is flushed.
Rsn	The reason for the graceful restart. Possible values: • UK – Unknown • RS – Software restart • UP – Software upgrade or reload • SW – Switch to redundant control processor
Nbr Intf IP	The IP address of the OSPF graceful restart neighbor.

Chapter

OSPF version 3 (IPv6) 6

Table 42 lists the Open Shortest Path First (OSPF) version 3 (IPv6) features Brocade ICX 6650 devices support. These features are supported with premium IPv6 devices running the full Layer 3 software image.

TABLE 42 Supported OSPF V3 features

Feature	Brocade ICX 6650
OSPF V3	Yes
Assigning OSPF V3 areas	Yes
Assigning interfaces to an area	Yes
Virtual links	Yes
Changing the reference bandwidth	Yes
Redistributing routes into OSPF V3	Yes
Filtering OSPF V3 routes	Yes
Configuring default route origination	Yes
Modifying SPF timers	Yes
Modifying administrative distance	Yes
OSPF V3 LSA pacing interval	Yes
Modifying the exit overflow interval	Yes
Modifying the external link state database limit	Yes
Modifying OSPF V3 interface defaults	Yes
Event logging	Yes
IPsec for OSPFv3	Yes

OSPF (IPv6) overview

Open Shortest Path First (OSPF) is a link-state routing protocol. OSPF uses link-state advertisements (LSAs) to update neighboring routers about its interfaces and information on those interfaces. The switch floods LSAs to all neighboring routers to update them about the interfaces. Each router maintains an identical database that describes its area topology to help a router determine the shortest path between it and any neighboring router:

- The differences between OSPF Version 2 (OSPF V2) and OSPF Version 3 (OSPF V3).
- The link state advertisement types for OSPF Version 3.
- How to configure OSPF Version 3.
- How to display OSPF Version 3 information and statistics.

NOTE
The terms *Layer 3 Switch* and *router* are used interchangeably in this chapter and mean the same thing.

Differences between OSPF V2 and OSPF V3

IPv6 supports OSPF V3 functions similarly to OSPF V2 (the current version that IPv4 supports), except for the following enhancements:

- Support for IPv6 addresses and prefixes.
- While OSPF V2 runs per IP subnet, OSPF V3 runs per link. In general, you can configure several IPv6 addresses on a router interface, but OSPF V3 forms one adjacency per interface only, using the interface associated link-local address as the source for OSPF protocol packets. On virtual links, OSPF V3 uses the global IP address as the source.
- You can run one instance of OSPF Version 2 and one instance of OSPF V3 concurrently on a link.
- Support for IPv6 link state advertisements (LSAs).

In addition, Brocade implements some new commands that are specific to OSPF V3. This chapter describes the commands that are specific to OSPF V3.

NOTE
Although OSPF Versions 2 and 3 function similarly to each other, Brocade has implemented the user interface for each version independently of each other. Therefore, any configuration of OSPF Version 2 features will not affect the configuration of OSPF V3 features and vice versa.

NOTE
You are required to configure a router ID when running only IPv6 routing protocols.

Link state advertisement types for OSPF V3

OSPF V3 supports the following types of LSAs:

- Router LSAs (Type 1)
- Network LSAs (Type 2)
- Interarea-prefix LSAs for ABRs (Type 3)
- Interarea-router LSAs for ASBRs (Type 4)
- Autonomous system external LSAs (Type 5)
- Link LSAs (Type 8)
- Intra-area prefix LSAs (Type 9)

For more information about these LSAs, see RFC 2740.

OSPF V3 configuration

To configure OSPF V3, you must perform the following tasks:

1. "Enabling OSPF V3" on page 229
2. "Assigning OSPF V3 areas" on page 230
3. "Assigning interfaces to an area" on page 231

The following configuration tasks are optional:

- Configure a virtual link between an ABR without a physical connection to a backbone area and the Brocade device in the same area with a physical connection to the backbone area.
- Change the reference bandwidth for the cost on OSPF V3 interfaces.
- Configure the redistribution of routes into OSPF V3.
- Configure default route origination.
- Modify the shortest path first (SPF) timers.
- Modify the administrative distances for OSPF V3 routes.
- Configure the OSPF V3 LSA pacing interval
- Modify how often the Brocade device checks on the elimination of the database overflow condition.
- Modify the external link state database limit.
- Modify the default values of OSPF V3 parameters for router interfaces.
- Disable or re-enable OSPF V3 event logging.

Enabling OSPF V3

Before enabling the Brocade device to run OSPF V3, you must do the following:

- Enable the forwarding of IPv6 traffic on the Brocade device using the **ipv6 unicast-routing** command.
- Enable IPv6 on each interface over which you plan to enable OSPF V3. You enable IPv6 on an interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

By default, OSPF V3 is disabled. To enable OSPF V3, you must enable it globally.

To enable OSPF V3 globally, enter the **ipv6 router ospf** command.

```
Brocade(config-ospf-router)#ipv6 router ospf
Brocade(config-ospf6-router)#
```

After you enter this command, the Brocade device enters the IPv6 OSPF configuration level, where you can access several commands that allow you to configure OSPF V3.

Syntax: [**no**] **ipv6 router ospf**

To disable OSPF V3, enter the **no** form of this command. If you disable OSPF V3, the Brocade device removes all the configuration information for the disabled protocol from the running-config. Moreover, when you save the configuration to the startup-config file after disabling one of these protocols, all the configuration information for the disabled protocol is removed from the startup-config file.

The CLI displays a warning message such as the following.

```
Brocade(config-ospf6-router)#no ipv6 router ospf
ipv6 router ospf mode now disabled. All ospf config data will be lost when writing
to flash!
```

If you have disabled the protocol but have not yet saved the configuration to the startup-config file and reloaded the software, you can restore the configuration information by re-entering the command to enable the protocol (for example, **ipv6 router ospf**). If you have already saved the configuration to the startup-config file and reloaded the software, the information is gone. If you are testing an OSPF configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

Assigning OSPF V3 areas

After OSPF V3 is enabled, you can assign OSPF V3 areas. You can assign an IPv4 address or a number as the ***area ID*** for each area. The area ID is representative of all IPv6 addresses (subnets) on a router interface. Each router interface can support one area.

An area can be **normal** or a **stub**:

- **Normal** – OSPF routers within a normal area can send and receive External Link State Advertisements (LSAs).
- **Stub** – OSPF routers within a stub area cannot send or receive External LSAs. In addition, OSPF routers in a stub area must use a default route to the area Area Border Router (ABR) or Autonomous System Boundary Router (ASBR) to send traffic out of the area.

For example, to set up OSPF V3 areas 0.0.0.0, 192.168.10.0, 192.168.1.0, and 192.168.0.0, enter the following commands.

```
Brocade(config-ospf6-router)#area 0.0.0.0
Brocade(config-ospf6-router)#area 192.168.10.0
Brocade(config-ospf6-router)#area 192.168.1.0
Brocade(config-ospf6-router)#area 192.168.0.0
```

Syntax: [**no**] **area** *number* | *ipv4-address*

The *number* | *ipv4-address* parameter specifies the area number, which can be a number or in IPv4 address format. If you specify a number, the number can be from 0 – 18.

NOTE
You can assign one area on a router interface.

Assigning a totally stubby area

By default, the Brocade device sends summary LSAs (LSA type 3) into stub areas. You can further reduce the number of LSAs sent into a stub area by configuring the Brocade device to stop sending summary LSAs into the area. You can disable the summary LSAs when you are configuring the stub area or later after you have configured the area.

This feature disables origination of summary LSAs into a stub area, but the Brocade device still accepts summary LSAs from OSPF neighbors and floods them to other areas. The Brocade device can form adjacencies with other routers regardless of whether summarization is enabled or disabled for areas on each router.

When you disable the summary LSAs, the change takes effect immediately. If you apply the option to a previously configured area, the router flushes all of the summary LSAs it has generated (as an ABR) from the area.

NOTE
This feature applies only when the Brocade device is configured as an Area Border Router (ABR) for the area. To completely prevent summary LSAs from being sent to the area, disable the summary LSAs on each OSPF router that is an ABR for the area.

For example, to disable summary LSAs for stub area 40 and specify an additional metric of 99, enter the following command.

```
Brocade(config-ospf6-router)#area 40 stub 99 no-summary
```

Syntax: **area** *number* | *ipv4-address* **stub** *metric* [**no-summary**]

The *number* | *ipv4-address* parameter specifies the area number, which can be a number or in IPv4 address format. If you specify a number, the number can be from 0–18.

The **stub** *metric* parameter specifies an additional cost for using a route to or from this area and can be from 1–16777215. There is no default. Normal areas do not use the cost parameter.

The **no-summary** parameter applies only to stub areas and disables summary LSAs from being sent into the area.

Assigning interfaces to an area

After you define OSPF V3 areas, you must assign router interfaces to the areas. All router interfaces must be assigned to one of the defined areas on an OSPF router. When an interface is assigned to an area, all corresponding subnets on that interface are automatically included in the assignment.

For example, to assign Ethernet interface 1/1/3 to area 192.168.0.0, enter the following commands.

```
Brocade(config)#interface ethernet 1/1/3
Brocade(config-if-e10000-1/1/3)#ipv6 ospf area 192.168.0.0
```

Syntax: [**no**] **ipv6 ospf area** *number* | *ipv4-address*

The *number* | *ipv4-address* parameter specifies the area number, which can be a number or in IPv4 address format. If you specify a number, the number can be from 0–18.

To remove the interface from the specified area, use the **no** form of this command.

Configuring virtual links

All ABRs must have either a direct or indirect link to an OSPF backbone area (0.0.0.0 or 0). If an ABR does not have a physical link to a backbone area, you can configure a virtual link from the ABR to another router within the same area that has a physical connection to the backbone area.

The path for a virtual link is through an area shared by the neighbor ABR (router with a physical backbone connection) and the ABR requiring a logical connection to the backbone.

Two parameters must be defined for all virtual links—transit area ID and neighbor router:

- The transit area ID represents the shared area of the two ABRs and serves as the connection point between the two routers. This number should match the area ID value.

- When assigned from the router interface requiring a logical connection, the neighbor router field is the router ID (IPv4 address) of the router that is physically connected to the backbone. When assigned from the router interface with the physical connection, the neighbor router is the router ID (IPv4) address of the router requiring a logical connection to the backbone.

NOTE
By default, the Brocade router ID is the IPv4 address configured on the lowest numbered loopback interface. If the Brocade device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device.

NOTE
When you establish an area virtual link, you must configure it on both of the routers (both ends of the virtual link).

For example, imagine that ABR1 in areas 1 and 2 is cut off from the backbone area (area 0). To provide backbone access to ABR1, you can add a virtual link between ABR1 and ABR2 in area 1 using area 1 as a transit area. To configure the virtual link, you define the link on the router that is at each end of the link. No configuration for the virtual link is required on the routers in the transit area.

To define the virtual link on ABR1, enter the following command on ABR1.

```
Brocade(config-ospf6-router)#area 1 virtual-link 192.168.22.1
```

To define the virtual link on ABR2, enter the following command on ABR2.

```
Brocade(config-ospf6-router)#area 1 virtual-link 10.0.0.1
```

Syntax: **area** *number* | *ipv4-address* **virtual-link** *router-id*

The **area** *number* | *ipv4-address* parameter specifies the transit area.

The *router-id* parameter specifies the router ID of the OSPF router at the remote end of the virtual link. To display the router ID on a router, enter the **show ip** command.

Assigning a virtual link source address

When routers at both ends of a virtual link need to communicate with one another, the source address included in the packets must be a global IPv6 address. Therefore, you must determine the global IPv6 address to be used by the routers for communication across the virtual link. You can specify that a router uses the IPv6 global address assigned to one of its interfaces.

For example, to specify the global IPv6 address assigned to Ethernet interface 1/1/3 on ABR1 as the source address for the virtual link on ABR1, enter the following command on ABR1.

```
Brocade(config-ospf6-router)#virtual-link-if-address interface ethernet 1/1/3
```

To specify the global IPv6 address assigned to tunnel interface 1 on ABR2 as the source address for the virtual link on ABR2, enter the following command on ABR2.

```
Brocade(config-ospf6-router)#virtual-link-if-address interface tunnel 1
```

Syntax: **virtual-link-if-address interface ethernet** *port* | **loopback** *number* | **tunnel** *number* | **ve** *number*

The **ethernet** | **loopback** | **tunnel** | **ve** parameter specifies the interface from which the router derives the source IPv6 address for communication across the virtual link. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a loopback, tunnel, or VE interface, also specify the number associated with the respective interface.

To delete the source address for the virtual link, use the **no** form of this command.

Modifying virtual link parameters

You can modify the following virtual link parameters:

- Dead-interval: The number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router is down. The range is 1 – 65535 seconds. The default is 40 seconds.
- Hello-interval: The length of time between the transmission of hello packets. The range is 1 – 65535 seconds. The default is 10 seconds.
- Retransmit-interval: The interval between the re-transmission of link state advertisements to router adjacencies for this interface. The range is 0 – 3600 seconds. The default is 5 seconds.
- Transmit-delay: The period of time it takes to transmit Link State Update packets on the interface. The range is 0 – 3600 seconds. The default is 1 second.

NOTE
The values of the **dead-interval** and **hello-interval** parameters must be the same at both ends of a virtual link. Therefore, if you modify the values of these parameters at one end of a virtual link, you must remember to make the same modifications on the other end of the link.

The values of the other virtual link parameters do not require synchronization.

For example, to change the dead interval to 60 seconds on the virtual links defined on ABR1 and ABR2, enter the following command on ABR1.

```
Brocade(config-ospf6-router)#area 1 virtual-link 192.168.22.1
dead-interval 60
```

Enter the following command on ABR2.

```
Brocade(config-ospf6-router)#area 1 virtual-link 10.0.0.1 dead-interval 60
```

Syntax: **area** *number* | *ipv4-address* **virtual-link** *router-id* [**dead-interval** *seconds* | **hello-interval** *seconds* | **retransmit-interval** *seconds* | **transmit-delay** *seconds*]

The **area** *number* | *ipv4-address* parameter specifies the transit area.

The *router-id* parameter specifies the router ID of the OSPF router at the remote end of the virtual link. To display the router ID on a router, enter the **show ip** command.

The **dead-interval**, **hello-interval**, **retransmit-interval**, and **transmit-delay** parameters are discussed earlier in this section.

Changing the reference bandwidth for the cost on OSPF V3 interfaces

Each interface on which OSPF V3 is enabled has a cost associated with it. The Brocade device advertises its interfaces and their costs to OSPF V3 neighbors. For example, if an interface has an OSPF cost of ten, the Brocade device advertises the interface with a cost of ten to other OSPF routers.

By default, an interface OSPF cost is based on the port speed of the interface. The software uses the following formula to calculate the cost.

Cost = reference-bandwidth/interface-speed

By default, the reference bandwidth is 100 Mbps. If the resulting cost is less than 1, the software rounds the cost up to 1. The default reference bandwidth results in the following costs:

- 10 Mbps port cost = 100/10 = 10
- 100 Mbps port cost = 100/100 = 1
- 1000 Mbps port cost = 100/1000 = 0.10, which is rounded up to 1
- 155 Mbps port cost = 100/155 = 0.65, which is rounded up to 1
- 622 Mbps port cost = 100/622 = 0.16, which is rounded up to 1
- 2488 Mbps port cost = 100/2488 = 0.04, which is rounded up to 1

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- Trunk group – The combined bandwidth of all the ports.
- Virtual (Ethernet) interface – The combined bandwidth of all the ports in the port-based VLAN that contains the virtual interface.

You can change the default reference bandwidth from 100 Mbps to a value from 1 – 4294967 Mbps.

If a change to the reference bandwidth results in a cost change to an interface, the Brocade device sends a link state update to update the costs of interfaces advertised by the Brocade device.

NOTE
If you specify the cost for an individual interface, the cost you specify overrides the cost calculated by the software.

Some interface types are not affected by the reference bandwidth and always have the same cost regardless of the reference bandwidth in use:

- The cost of a loopback interface is always 0.
- The cost of a virtual link is calculated using the Shortest Path First (SPF) algorithm and is not affected by the auto-cost feature.

For example, to change the reference bandwidth to 500, enter the following command.

```
Brocade(config-ospf6-router)#auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port cost = 500/10 = 50
- 100 Mbps port cost = 500/100 = 5
- 1000 Mbps port cost = 500/1000 = 0.5, which is rounded up to 1

- 155 Mbps port cost = 500/155 = 3.23, which is rounded up to 4
- 622 Mbps port cost = 500/622 = 0.80, which is rounded up to 1
- 2488 Mbps port cost = 500/2488 = 0.20, which is rounded up to 1

The costs for 10 Mbps, 100 Mbps, and 155 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

Syntax: [**no**] **auto-cost reference-bandwidth** *number*

The *number* parameter specifies the reference bandwidth and can be a value from 1 – 4294967. The default is 100.

To restore the reference bandwidth to its default value and thus restore the default costs of interfaces to their default values, enter the **no** form of this command.

Redistributing routes into OSPF V3

In addition to specifying which routes are redistributed into OSPF V3, you can configure the following aspects related to route redistribution:

- Default metric
- Metric type
- Advertisement of an external aggregate route

Configuring route redistribution into OSPF V3

You can configure the Brocade device to redistribute routes from the following sources into OSPF V3:

- IPv6 static routes
- Directly connected IPv6 networks
- RIPng

You can redistribute routes in the following ways:

- By route types, for example, the Brocade device redistributes all IPv6 static and RIPng routes.
- By using a route map to filter which routes to redistribute, for example, the Brocade device redistributes specified IPv6 static and RIPng routes only.

For example, to configure the redistribution of all IPv6 static RIPng routes, enter the following commands.

```
Brocade(config-ospf6-router)#redistribute static
Brocade(config-ospf6-router)#redistribute rip
```

Syntax: [**no**] **redistribute bgp** | **connected** | **rip** | **static** [**metric** *number* | **metric-type** *type*]

The **connected** | **rip** | **static** keywords specify the route source.

The **metric** *number* parameter specifies the metric used for the redistributed route. If a value is not specified for this option, and the value for the **default-metric** command is set to 0, its default metric, then routes redistributed from the various routing protocols will have the metric value of the protocol from which they are redistributed. For information about the **default-metric** command, refer to "Modifying default metric for routes redistributed into OSPF V3" on page 237

The **metric-type** *type* parameter specifies an OSPF metric type for the redistributed route. You can specify external type 1 or external type 2. If a value is not specified for this option, the Brocade device uses the value specified by the **metric-type** command. For information about modifying the default metric type using the **metric-type** command, refer to "Modifying default metric for routes redistributed into OSPF V3" on page 237

For example, to configure a route map and use it for redistribution of routes into OSPF V3, enter commands such as the following.

```
Brocade(config)#ipv6 route 2001:db8::/32 0000:00ff:343e::23
Brocade(config)#ipv6 route 2001:db8::/32 0000:00ff:343e::23
Brocade(config)#ipv6 route 2001:db8::/32 0000:00ff:343e::23 metric 5
Brocade(config)#route-map abc permit 1
Brocade(config-routemap abc)#match metric 5
Brocade(config-routemap abc)#set metric 8
Brocade(config-routemap abc)#ipv6 router ospf
Brocade(config-ospf6-router)#redistribute static route-map abc
```

The commands in this example configure some static IPv6 routes and a route map, and use the route map for redistributing the static IPv6 routes into OSPF V3.

The **ipv6 route** commands configure the static IPv6 routes.

The **route-map** command begins configuration of a route map called "abc". The number indicates the route map entry (called the "instance") you are configuring. A route map can contain multiple entries. The software compares packets to the route map entries in ascending numerical order and stops the comparison once a match is found.

NOTE
The default action rule for route-map is to deny all routes that are not explicitly permitted. Refer to "Configuring an OSPF V3 distribution list using a route map that uses a prefix list" on page 241.

The **match** command in the route map matches on routes that have 5 for their metric value (cost). The **set** command changes the metric in routes that match the route map to 8.

The **redistribute** command configures the redistribution of static IPv6 routes into OSPF V3, and uses route map "abc" to control the routes that are redistributed. In this example, the route map allows a static IPv6 route to be redistributed into OSPF only if the route has a metric of 5, and changes the metric to 8 before placing the route into the OSPF route redistribution table.

Syntax: [no] **redistribute bgp** | **connected** | **isis** | **rip** | **static** [**route-map** *map-name*]

The **bgp** | **connected** | **isis** | **rip** | **static** keywords specify the route source.

The **route-map** *map-name* parameter specifies the route map name. The following match parameters are valid for OSPF V3 redistribution:

- **match metric** *number*

The following set parameters are valid for OSPF redistribution:

- **set metric** [+ | -] *number* | none
- **set metric-type type-1** | **type-2**

NOTE
You must configure the route map before you configure a redistribution filter that uses the route map.

NOTE
When you use a route map for route redistribution, the software disregards the permit or deny action of the route map.

NOTE
For an external route that is redistributed into OSPF V3 through a route map, the metric value of the route remains the same unless the metric is set by a **set metric** command inside the route map or the **default-metric** *num* command. For a route redistributed without using a route map, the metric is set by the metric parameter if set or the **default-metric** *num* command if the metric parameter is not set.

Modifying default metric for routes redistributed into OSPF V3

The default metric is a global parameter that specifies the cost applied by default to routes redistributed into OSPF V3. The default value is 0.

If the **metric** parameter for the **redistribute** command is not set and the **default-metric** command is set to 0, its default value, then routes redistributed from the various routing protocols will have the metric value of the protocol from which they are redistributed. For information about the **redistribute** command, refer to “Configuring route redistribution into OSPF V3” on page 235.

NOTE
You also can define the cost on individual interfaces. The interface cost overrides the default cost. For information about defining the cost on individual interfaces, refer to “Modifying OSPF V3 interface defaults” on page 246 and “Changing the reference bandwidth for the cost on OSPF V3 interfaces” on page 234.

To assign a default metric of 4 to all routes imported into OSPF V3, enter the **default-metric** command.

```
Brocade(config-ospf6-router)#default-metric 4
```

Syntax: no] **default-metric** *number*

You can specify a value from 0 – 65535. The default is 0.

To restore the default metric to the default value, use the **no** form of this command.

Modifying metric type for routes redistributed into OSPF V3

The Brocade device uses the **metric-type** parameter by default for all routes redistributed into OSPF V3 unless you specify a different metric type for individual routes using the **redistribute** command. (For more information about using the **redistribute** command, refer to “Redistributing routes into OSPF V3” on page 235).

A type 1 route specifies a small metric (two bytes), while a type 2 route specifies a big metric (three bytes). The default value is type 2.

To modify the default value of type 2 to type 1, enter the **metric-type** command.

```
Brocade(config-ospf6-router)#metric-type type1
```

Syntax: no] **metric-type type1** | **type2**

To restore the metric type to the default value, use the **no** form of this command.

External route summarization

When the Brocade device is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to advertise one external route as an aggregate for all redistributed routes that are covered by a specified IPv6 address range.

When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the Brocade device, no action is taken if the device has already advertised the aggregate route; otherwise, the device advertises the aggregate route. If an imported route that falls within a configured address range is removed by the device, no action is taken if there are other imported routes that fall with in the same address range; otherwise the aggregate route is flushed.

You can configure up to 32 address ranges. The Brocade device sets the forwarding address of the aggregate route to zero and sets the tag to zero.

If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually.

If an external link state database overflow (LSDB) condition occurs, all aggregate routes are flushed out of the AS, along with other external routes. When the Brocade device exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.

NOTE
If you use redistribution filters in addition to address ranges, the Brocade device applies the redistribution filters to routes first, then applies them to the address ranges.

NOTE
If you disable redistribution, all the aggregate routes are flushed, along with other imported routes.

NOTE
This option affects only imported, type 5 external routes. A single type 5 LSA is generated and flooded throughout the AS for multiple external routes.

Configuring external route summarization

To configure the summary address 2001:db8::/24 for routes redistributed into OSPF V3, enter the following command.

```
Brocade(config-ospf6-router)#summary-address 2001:db8::/24
```

In this example, the summary prefix 2001:db8::/24 includes addresses 2001:db8::/1 through 2001:db8::/24. Only the address FEC0::/24 is advertised in an external link-state advertisement.

Syntax: **summary-address** *ipv6-prefix*/*prefix-length*

You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

Filtering OSPF V3 routes

You can filter the routes to be placed in the OSPF V3 route table by configuring distribution lists. OSPF V3 distribution lists can be applied globally or to an interface.

The functionality of OSPF V3 distribution lists is similar to that of OSPFv2 distribution lists. However, unlike OSPFv2 distribution lists, which filter routes based on criteria specified in an Access Control List (ACL), OSPF V3 distribution lists can filter routes using information specified in an IPv6 prefix list or a route map.

Configuration examples for filtering OSPF V3 routes

The following sections show examples of filtering OSPF V3 routes using prefix lists globally and for a specific interface, as well as filtering OSPF V3 routes using a route map.

You can configure the device to use all three types of filtering. When you do this, filtering using route maps has higher priority over filtering using global prefix lists. Filtering using prefix lists for a specific interface has lower priority than the other two filtering methods.

The example in this section assume the following routes are in the OSPF V3 route table.

```
Brocade#show ipv6 ospf route

  Current Route count: 5
    Intra: 3 Inter: 0 External: 2 (Type1 0/Type2 2)
    Equal-cost multi-path: 0
Destination                    Options   Area         Cost Type2 Cost
    Next Hop Router                Outgoing Interface
*IA 2001:db8::/64                  --------- 0.0.0.1        0        0
    ::                             ve 10
*E2 2001:db8::/64                  --------- 0.0.0.0        10       0
    2001:db8:2e0:52ff:fe00:10      ve 10
*IA 2001:db8::/64                  V6E---R-- 0.0.0.0        11       0
    2001:db8:2e0:52ff:fe00:10      ve 10
*IA 2001:db8::/64                  --------- 0.0.0.0        10       0
    ::                             ve 11
*E2 2001:db8::/64                  --------- 0.0.0.0        10       0
    2001:db8:2e0:52ff:fe00:10      ve 10
```

Configuring an OSPF V3 distribution list using an IPv6 prefix list as input

The following example illustrates how to use an IPv6 prefix list is used to filter OSPF V3 routes.

To specify an IPv6 prefix list called filterOspfRoutes that denies route 2001:db8::/64, enter the following commands.

```
Brocade(config)#ipv6 prefix-list  filterOspfRoutes seq 5 deny 2001:db8::/64
Brocade(config)#ipv6 prefix-list  filterOspfRoutes seq 7 permit ::/0 ge 1 le 128
```

Syntax: **ipv6 prefix-list** *name* [**seq** *seq-value*] [**description** *string*] **deny** | **permit** *ipv6-addr/mask-bits* [**ge** *ge-value*] [**le** *le-value*]

To configure a distribution list that applies the filterOspfRoutes prefix list globally.

```
Brocade(config)#ipv6 router ospf
Brocade(config-ospf6-router)#distribute-list prefix-list filterOspfRoutes in
```

Syntax: [**no**] **distribute-list prefix-list** *name* **in** [*interface*]

After this distribution list is configured, route 2001:db8::/64 would be omitted from the OSPF V3 route table.

```
Brocade#show ipv6 ospf route

  Current Route count: 4
    Intra: 3 Inter: 0 External: 1 (Type1 0/Type2 1)
    Equal-cost multi-path: 0
Destination                    Options   Area             Cost Type2 Cost
    Next Hop Router                Outgoing Interface
*IA 2001:db8::/64                  --------- 0.0.0.1             0   0
    ::                             ve 10
*IA 2001:db8::/64                  V6E---R-- 0.0.0.0            11   0
    2001:db8:2e0:52ff:fe00:10      ve 10
*IA 2001:db8::/64                  --------- 0.0.0.0            10   0
    ::                             ve 11
*E2 2001:db8::/64                  --------- 0.0.0.0            10   0
    2001:db8:2e0:52ff:fe00:10      ve 10
```

The following commands specify an IPv6 prefix list called filterOspfRoutesVe that denies route 2001:db8::/64.

```
Brocade(config)#ipv6 prefix-list filterOspfRoutesVe seq 5 deny 2001:db8::/64
Brocade(config)#ipv6 prefix-list filterOspfRoutesVe seq 10 permit ::/0 ge 1 le 128
```

The following commands configure a distribution list that applies the filterOspfRoutesVe prefix list to routes pointing to virtual interface 10.

```
Brocade(config)#ipv6 router ospf
Brocade(config-ospf6-router)#distribute-list prefix-list filterOspfRoutes in ve
10
```

After this distribution list is configured, route 2001:db8::/64, pointing to virtual interface 10, would be omitted from the OSPF V3 route table.

```
Brocade#show ipv6 ospf route

  Current Route count: 4
    Intra: 3 Inter: 0 External: 1 (Type1 0/Type2 1)
    Equal-cost multi-path: 0
Destination                    Options   Area             Cost Type2 Cost
    Next Hop Router                Outgoing Interface
*IA 2001:db8::/64                  --------- 0.0.0.1             0   0
    ::                             ve 10
*E2 2001:db8::/64                  --------- 0.0.0.0            10   0
    2001:db8:2e0:52ff:fe00:10      ve 10
*IA 2001:db8::/64                  --------- 0.0.0.0            10   0
    ::                             ve 11
*E2 2001:db8::/64                  --------- 0.0.0.0            10   0
    2001:db8:2e0:52ff:fe00:10      ve 10
```

Configuring an OSPF V3 distribution list using a route map as input

The following commands configure a route map that matches internal routes.

```
Brocade(config)#route-map allowInternalRoutes permit 10
Brocade(config-routemap allowInternalRoutes)#match route-type internal
```

Refer to “Policy-Based Routing” for information on configuring route maps.

The following commands configure a distribution list that applies the allowInternalRoutes route map globally to OSPF V3 routes.

```
Brocade(config)#ipv6 router ospf
Brocade(config-ospf6-router)#distribute-list route-map allowinternalroutes in
```

Syntax: [no] distribute-list route-map *name* in

After this distribution list is configured, the internal routes would be included, and the external routes would be omitted from the OSPF V3 route table.

```
Brocade#show ipv6 ospf route

  Current Route count: 3
    Intra: 3 Inter: 0 External: 0 (Type1 0/Type2 0)
    Equal-cost multi-path: 0
Destination                     Options   Area            Cost Type2 Cost
    Next Hop Router                 Outgoing Interface
*IA 2001:db8::/64                   --------- 0.0.0.1            0   0
    ::                              ve 10
*IA 2001:db8::/64                   V6E---R-- 0.0.0.0           11   0
    2001:db8:2e0:52ff:fe00:10       ve 10
*IA 2001:db8::/64                   --------- 0.0.0.0           10   0
    ::                              ve 11
```

Configuring an OSPF V3 distribution list using a route map that uses a prefix list

When you configure route redistribution into OSPF V3 using a route map that uses a prefix list, the device supports both **permit** and **deny** statements in the route map and **permit** statements only in the prefix list. Therefore, the action to permit or deny is determined by the route map, and the conditions for the action are contained in the prefix list. The following shows an example configuration.

```
Brocade(config)#route-map v64 deny 10
Brocade(config-routemap v64)#match ipv6 next-hop prefix-list ospf-filter5
Brocade(config-routemap v64)#route-map v64 deny 11
Brocade(config-routemap v64)#match ipv6 address prefix-list ospf-filter2
Brocade(config-routemap v64)#route-map v64 permit 12
Brocade(config-routemap v64)#exit
Brocade(config)#ipv6 prefix-list ospf-filter2 seq 15 permit
2001:db8:2001:102::/64 ge 65 le 96
Brocade(config)#ipv6 prefix-list ospf-filter5 seq 15 permit
2001:db8:2e0:52ff:fe00:100/128
```

In this example the prefix lists, **ospf-filter2** and **ospf-filter5**, contain a range of IPv6 routes and one host route to be denied, and the route map **v64** defines the deny action.

NOTE
The default action rule for **route-map** is to deny all routes that are not explicitly permitted. If you configure a “deny” route map but want to permit other routes that do not match the rule, configure an “empty” permit route map. For example.

```
Brocade(config)#route-map abc deny 10
Brocade(config-routemap abc)#match metric 20
Brocade(config-routemap abc)#route-map abc permit 20
```

Without the last line in the above example, all routes would be denied.

Default route origination

When the Brocade device is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to automatically generate a default external route into an OSPF V3 routing domain. This feature is called “default route origination” or “default information origination.”

By default, the Brocade device does not advertise the default route into the OSPF V3 domain. If you want the device to advertise the OSPF default route, you must explicitly enable default route origination.

When you enable OSPF default route origination, the device advertises a type 5 default route that is flooded throughout the AS (except stub areas).

The device advertises the default route into OSPF even if OSPF route redistribution is not enabled, and even if the default route is learned through an IBGP neighbor.

NOTE
The Brocade device does not advertise the OSPF default route, regardless of other configuration parameters, unless you explicitly enable default route origination.

If default route origination is enabled and you disable it, the default route originated by the device is flushed. Default routes generated by other OSPF routers are not affected. If you re-enable the feature, the feature takes effect immediately and thus does not require you to reload the software.

Configuring a default route origination

To create and advertise a default route with a metric of 2 and as a type 1 external route, enter the following command.

```
Brocade(config-ospf6-router)#default-information-originate always metric 2
metric-type type1
```

Syntax: [**no**] **default-information-originate** [**always**] [**metric** *value*] [**metric-type** *type*]

The **always** keyword originates a default route regardless of whether the device has learned a default route. This option is disabled by default.

The **metric** *value* parameter specifies a metric for the default route. If this option is not used, the value of the **default-metric** command is used for the route. For information about this command, refer to “Modifying default metric for routes redistributed into OSPF V3” on page 237

The **metric-type** *type* parameter specifies the external link type associated with the default route advertised into the OSPF routing domain. The *type* can be one of the following:

- 1 – Type 1 external route
- 2 – Type 2 external route

If you do not use this option, the default redistribution metric type is used for the route type.

NOTE
If you specify a metric and metric type, the values you specify are used even if you do not use the always option.

To disable default route origination, enter the **no** form of the command.

Shortest path first timers

The Brocade device uses the following timers when calculating the shortest path for OSPF V3 routes:

- **SPF delay** – When the Brocade device receives a topology change, the software waits before it starts a Shortest Path First (SPF) calculation. By default, the software waits 5 seconds. You can configure the SPF delay to a value from 0 – 65535 seconds. If you set the SPF delay to 0 seconds, the software immediately begins the SPF calculation after receiving a topology change.
- **SPF hold time** – The Brocade device waits a specific amount of time between consecutive SPF calculations. By default, the device waits 10 seconds. You can configure the SPF hold time to a value from 0 – 65535 seconds. If you set the SPF hold time to 0 seconds, the software does not wait between consecutive SPF calculations.

You can set the SPF delay and hold time to lower values to cause the device to change to alternate paths more quickly if a route fails. Note that lower values for these parameters require more CPU processing time.

You can change one or both of the timers.

NOTE
If you want to change only one of the timers, for example, the SPF delay timer, you must specify the new value for this timer as well as the current value of the SPF hold timer, which you want to retain. The Brocade device does not accept only one timer value.

Modifying shortest path first timers

To change the SPF delay to 10 seconds and the SPF hold to 20 seconds, enter the following command.

```
Brocade(config-ospf6-router)#timers spf 10 20
```

Syntax: **timers spf** *delay hold-time*

For the *delay* and *hold-time* parameters, specify a value from 0–65535 seconds.

To set the timers back to their default values, enter the **no** version of this command.

Administrative distance

The Brocade device can learn about networks from various protocols, including IPv6, RIPng, and OSPF V3. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned. By default, the administrative distance for OSPF V3 routes is 110.

The device selects one route over another based on the source of the route information. To do so, the device can use the administrative distances assigned to the sources. You can influence the device decision by changing the default administrative distance for OSPF V3 routes.

Configuring administrative distance based on route type

You can configure a unique administrative distance for each type of OSPF V3 route. For example, you can use this feature to influence the Brocade device to prefer a static route over an OSPF inter-area route and to prefer OSPF intra-area routes to static routes.

The distance you specify influences the choice of routes when the device has multiple routes to the same network from different protocols. The device prefers the route with the lower administrative distance.

You can specify unique default administrative distances for the following OSPF V3 route types:

- Intra-area routes
- Inter-area routes
- External routes

The default for all of these OSPF V3 route types is 110.

NOTE
This feature does not influence the choice of routes within OSPF V3. For example, an OSPF intra-area route is always preferred over an OSPF inter-area route, even if the intra-area route distance is greater than the inter-area route distance.

For example, to change the default administrative distances for intra-area routes to 80, inter-area routes to 90, and external routes to 100, enter the following commands.

```
Brocade(config-ospf6-router)#distance intra-area 80
Brocade(config-ospf6-router)#distance inter-area 90
Brocade(config-ospf6-router)#distance external 100
```

Syntax: **distance external** | **inter-area** | **intra-area** *distance*

The **external** | **inter-area** | **intra-area** keywords specify the route type for which you are changing the default administrative distance.

The *distance* parameter specifies the new distance for the specified route type. You can specify a value from 1–255.

To reset the administrative distance of a route type to its system default, enter the **no** form of this command.

Configuring the OSPF V3 LSA pacing interval

The Brocade device paces OSPF V3 LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the Brocade device refreshes and sends out together in one or more packets.

The pacing interval, which is the interval at which the Brocade device refreshes an accumulated group of LSAs, is configurable to a range from 10–1800 seconds (30 minutes). The default is 240 seconds (four minutes). Thus, every four minutes, the Brocade device refreshes the group of accumulated LSAs and sends the group together in the same packets.

The pacing interval is inversely proportional to the number of LSAs the Brocade device is refreshing and aging. For example, if you have approximately 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a very small database (40–100 LSAs), increasing the pacing interval to 10–20 minutes might enhance performance only slightly.

To change the OSPF V3 LSA pacing interval to two minutes (120 seconds), enter the following command.

```
Brocade(config)#ipv6 router ospf
Brocade(config-ospf6-router)#timers lsa-group-pacing 120
```

Syntax: [**no**] **timers lsa-group-pacing** *seconds*

The *seconds* parameter specifies the number of seconds and can be from 10–1800 (30 minutes). The default is 240 seconds (four minutes).

To restore the pacing interval to its default value, use the **no** form of the command.

Modifying exit overflow interval

If a database overflow condition occurs on the Brocade device, the device eliminates the condition by removing entries that originated on the device. The exit overflow interval allows you to set how often a device checks to see if the overflow condition has been eliminated. The default value is 0. If the configured value of the database overflow interval is 0, then the device never leaves the database overflow condition.

For example, to modify the exit overflow interval to 60 seconds, enter the following command.

```
Brocade(config-ospf6-router)#database-overflow-interval 60
```

Syntax: [**no**] **auto-cost reference-bandwidth** *number*

The *seconds* parameter can be a value from 0–86400 seconds (24 hours).

To reset the exit overflow interval to its system default, enter the **no** form of this command.

Modifying external link state database limit

By default, the link state database can hold a maximum of 2000 entries for external (type 5) LSAs. You can change the maximum number of entries from 500–8000. After changing this limit, make sure to save the running-config file and reload the software. The change does not take effect until you reload or reboot the software.

For example, to change the maximum number entries from the default of 2000 to 3000, enter the following command.

```
Brocade(config-ospf6-router)#external-lsdb-limit 3000
```

Syntax: **ipv6 ospf area** *number* | *ipv4-address*

The *entries* parameter can be a numerical value from 500–8000 seconds.

To reset the maximum number of entries to its system default, enter the **no** form of this command.

Modifying OSPF V3 interface defaults

OSPF V3 has interface parameters that you can configure. For simplicity, each of these parameters has a default value. No change to these default values is required except as needed for specific network configurations.

You can modify the default values for the following OSPF interface parameters:

- **Cost**: Indicates the overhead required to send a packet across an interface. You can modify the cost to differentiate between 100 Mbps and 1000 Mbps (1 Gbps) links. The command syntax is **ipv6 ospf cost** *number*. The default cost is calculated by dividing 100 million by the bandwidth. For 10 Mbps links, the cost is 10. The cost for both 100 Mbps and 1000 Mbps links is 1, because the speed of 1000 Mbps was not in use at the time the OSPF cost formula was devised.
- **Dead-interval**: Indicates the number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router down. The command syntax is **ipv6 ospf dead-interval** *seconds*. The value can be from 1–2147483647 seconds. The default is 40 seconds.
- **Hello-interval:** Represents the length of time between the transmission of hello packets. The command syntax is **ipv6 ospf hello-interval** *seconds*. The value can be from 1 – 65535 seconds. The default is 10 seconds.
- **Instance:** Indicates the number of OSPF V3 instances running on an interface. The command syntax is **ipv6 ospf instance** *number*. The value can be from 0–255. The default is 1.
- **MTU-ignore:** Allows you to disable a check that verifies the same MTU is used on an interface shared by neighbors. The command syntax is **ipv6 ospf mtu-ignore**. By default, the mismatch detection is enabled.
- **Network:** Allows you to configure the OSPF network type. The command syntax is **ipv6 ospf network** [**point-to-multipoint**]. The default setting of the parameter depends on the network type.
- **Passive**: When you configure an OSPF interface to be passive, that interface does not send or receive OSPF route updates. This option affects all IPv6 subnets configured on the interface. The command syntax is **ipv6 ospf passive.** By default, all OSPF interfaces are active and thus can send and receive OSPF route information. Since a passive interface does not send or receive route information, the interface is in effect a stub network.
- **Priority:** Allows you to modify the priority of an OSPF router. The priority is used when selecting the designated router (DR) and backup designated routers (BDRs). The command syntax is **ipv6 ospf priority** *number*. The value can be from 0–255. The default is 1. If you set the priority to 0, the router does not participate in DR and BDR election.
- **Retransmit-interval:** The time between retransmissions of LSAs to adjacent routers for an interface. The command syntax is **ipv6 ospf retransmit-interval** *seconds*. The value can be from 0–3600 seconds. The default is 5 seconds.

- **Transmit-delay:** The time it takes to transmit Link State Update packets on this interface. The command syntax is **ipv6 ospf transmit-delay** *seconds*. The value can be from 0–3600 seconds. The default is 1 second.

Disabling or re-enabling event logging

OSPF V3 does not currently support the generation of SNMP traps. Instead, you can disable or re-enable the logging of OSPF V3-related events such as neighbor state changes and database overflow conditions. By default, the Brocade device logs these events. Since the OSPFv3 logs are enabled by defaut, the following log messages appear once the system is up.

```
May  8 10:06:09:N:OSPFv3 originate LSA, rid 10.16.16.16, area 0.0.0.16, LSA type
IntraPrefix, LSA id 0.0.0.10, LSA router id 10.16.16.16
May  8 10:06:09:N:OSPFv3 originate LSA, rid 10.16.16.16, area 0.0.0.16, LSA type
Network, LSA id 0.0.0.2, LSA router id 10.16.16.16
May  8 10:06:08:N:OSPFv3 originate LSA, rid 10.16.16.16, area 0.0.0.16, LSA type
Router, LSA id 0.0.0.0, LSA router id 10.16.16.16
```

To disable the logging of events, enter the following command.

```
Brocade(config-ospf6-router)#no log-status-change
```

Syntax: [**no**] **log-status-change**

To re-enable the logging of events, enter the following command.

```
Brocade(config-ospf6-router)#log-status-change
```

IPsec for OSPF V3

This section describes the implementation of Internet Protocol Security (IPsec) for securing OSPFv3 traffic. For background information and configuration steps, refer to "IPsec for OSPF V3 configuration" on page 248.

IPsec is available for OSPFv3 traffic only and only for packets that are "for-us." A for-us packet is addressed to one of the IPv6 addresses on the device or to an IPv6 multicast address. Packets that are just forwarded by the line card do not receive IPsec scrutiny.

Brocade devices support the following components of IPsec for IPv6-addressed packets:

- Authentication through Encapsulating Security Payload (ESP) in transport mode
- HMAC-SHA1-96 as the authentication algorithm
- Manual configuration of keys
- Configurable rollover timer

IPsec can be enabled on the following logical entities:

- Interface
- Area
- Virtual link

With respect to traffic classes, this implementation of IPsec uses a single security association (SA) between the source and destination to support all traffic classes and so does not differentiate between the different classes of traffic that the DSCP bits define.

Instructions for configuring IPsec on these entities appear in "IPsec for OSPF V3 configuration" on page 248.

IPsec on a virtual link is a global configuration. Interface and area IPsec configurations are more granular.

Among the entities that can have IPsec protection, the interfaces and areas can overlap. The interface IPsec configuration takes precedence over the area IPsec configuration when an area and an interface within that area use IPsec. Therefore, if you configure IPsec for an interface and an area configuration also exists that includes this interface, the interface's IPsec configuration is used by that interface. However, if you disable IPsec on an interface, IPsec is disabled on the interface even if the interface has its own, specific authentication. Refer to "Disabling IPsec on an interface" on page 253.

For IPsec, the system generates two types of databases. The *security association database* (SAD) contains a security association for each interface or one global database for a virtual link. Even if IPsec is configured for an area, each interface that uses the area's IPsec still has its own security association in the SAD. Each SA in the SAD is a generated entry that is based on your specifications of an authentication protocol (ESP in the current release), destination address, and a security policy index (SPI). The SPI number is user-specified according to the network plan. Consideration for the SPI values to specify must apply to the whole network.

The system-generated security policy databases (SPDs) contain the security policies against which the system checks the for-us packets. For each for-us packet that has an ESP header, the applicable security policy in the security policy database (SPD) is checked to see if this packet complies with the policy. The IPsec task drops the non-compliant packets. Compliant packets continue on to the OSPFv3 module.

IPsec for OSPF V3 configuration

This section describes how to configure IPsec for an interface, area, and virtual link. It also describes how to change the key rollover timer if necessary and how to disable IPsec on a particular interface for special purposes.

By default, OSPFv3 IPsec authentication is disabled. The following IPsec parameters are configurable:

- ESP security protocol
- Authentication
- HMAC-SHA1-96 authentication algorithm
- Security parameter index (SPI)
- A 40-character key using hexadecimal characters
- An option for not encrypting the keyword when it appears in **show** command output
- Key rollover timer

NOTE
In the current release, certain keyword parameters must be entered even though only one keyword choice is possible for that parameter. For example, the only authentication algorithm in the current release is HMAC-SHA1-96, but you must nevertheless enter the keyword for this algorithm. Also, ESP currently is the only authentication protocol, but you must still enter the **esp** keyword. This section describes all keywords.

General considerations when configuring IPsec for OSPF V3

The IPsec component generates security associations and security policies based on certain user-specified parameters. The parameters are described with the syntax of each command in this section and also pointed out in the section with the **show** command examples, "IPsec examples" on page 274. User-specified parameters and their relation to system-generated values are as follows:

- **Security association:** based on your entries for *security policy index* (SPI), *destination address*, and *security protocol* (currently ESP), the system creates a security association for each interface or virtual link.
- **Security policy database:** based on your entries for SPI, *source address*, *destination addresses*, and *security protocol*, the system creates a security policy database for each interface or virtual link.
- You can configure the same SPI and key on multiple interfaces and areas, but they still have unique IPsec configurations because the SA and policies are added to each separate security policy database (SPD) that is associated with a particular interface. If you configure an SA with the same SPI in multiple places, the rest of the parameters associated with the SA—such as key, crypto algorithm, and security protocol, and so on—must match. If the system detects a mismatch, it displays an error message.
- IPsec authentication for OSPFv3 requires the use of multiple SPDs, one for each interface. A virtual link has a separate, global SPD. The authentication configuration on a virtual link must be different from the authentication configuration for an area or interface, as required by RFC4552. The interface number is used to generate a non-zero security policy database identifier (SPDID), but for the global SPD for a virtual link, the system-generated SPDID is always zero. As a hypothetical example, the SPD for interface eth 1/1/1 might have the system-generated SPDID of 1, and so on.
- If you change an existing key, you must also specify a different SPI value. For example, in an interface context where you intend to change a key, you must type a different SPI value—which occurs before the key parameter on the command line—before you type the new key. The example in "IPsec for OSPF V3 configuration"illustrates this requirement.
- The old key is active for twice the current configured key-rollover-interval for the inbound direction. In the outbound direction, the old key remains active for a duration equal to the key-rollover-interval. If the key-rollover-interval is set to 0, the new key immediately takes effect for both directions. For a description of the key-rollover-interval, refer to the "Changing the key rollover timer" on page 254section.

Interface and area IPsec considerations

This section describes the precedence of interface and area IPsec configurations.

If you configure an interface IPsec by using the **ipv6 ospf authentication** command in the context of a specific interface, that interface's IPsec configuration overrides the area configuration of IPsec.

If you configure IPsec for an area, all interfaces that utilize the area-wide IPsec (where interface-specific IPsec is not configured) nevertheles receive an SPD entry (and SPDID number) that is unique for the interface.

The area-wide SPI that you specify is a constant for all interfaces in the area that use the area IPsec, but the use of different interfaces results in an SPDID and an SA that are unique to each interface. (Recall from "IPsec for OSPF V3" on page 247 that the security policy database depends partly on the source IP address, so a unique SPD for each interface results.)

Considerations for IPsec on virtual links

The IPsec configuration for a virtual link is global, so only one security association database and one security policy database exist for virtual links if you choose to configure IPsec for virtual links.

The virtual link IPsec SAs and policies are added to all interfaces of the transit area for the outbound direction. For the inbound direction, IPsec SAs and policies for virtual links are added to the global database.

NOTE
The security association (SA), security protocol index (SPI), security protocol database (SPD), and key have mutual dependencies, as the subsections that follow describe.

Specifying the key rollover timer

Configuration changes for authentication takes effect in a controlled manner through the key rollover procedure as specified in RFC 4552, Section 10.1. The key rollover timer controls the timing of the configuration changeover. The key rollover timer can be configured in the IPv6 router OSPF context, as the following example illustrates.

```
Brocade(config-ospf6-router)#key-rollover-interval 200
```

Syntax: **key-rollover-interval** *time*

The range for the key-rollover-interval is 0–14400 seconds. The default is 300 seconds.

Configuring IPsec on a interface

For IPsec to work, the IPsec configuration must be the same on all the routers to which an interface connects.

For multicast, IPsec does not need or use a specific destination address—the destination address is "do not care," and this status is reflected by the lone pair of colons (::) for destination address in the **show** command output.

To configure IPsec on an interface, proceed as in the following example.

NOTE
The IPsec configuration for an interface applies to the inbound and outbound directions. Also, the same authentication parameters must be used by all routers on the network to which the interface is connected, as described in section 7 of RFC 4552.

```
Brocade(config-if-e10000-1/1/2)#ipv6 ospf auth ipsec spi 429496795 esp sha1
abcdef12345678900987654321fedcba12345678
```

Syntax: [no] **ipv6 ospf authentication ipsec spi** *spinum* **esp sha1** [**no-encrypt**] *key*

The **no** form of this command deletes IPsec from the interface.

The **ipv6** command is available in the configuration interface context for a specific interface.

The **ospf** keyword identifies OSPFv3 as the protocol to receive IPsec security.

The **authentication** keyword enables authentication.

The **ipsec** keyword specifies IPsec as the authentication protocol.

The **spi** keyword and the *spinum* variable specify the security parameter that points to the security association. The near-end and far-end values for spinum must be the same. The range for *spinum* is decimal 256–4294967295.

The mandatory **esp** keyword specifies ESP (rather than authentication header) as the protocol to provide packet-level security. In the current release, this parameter can be **esp** only.

The **sha1** keyword specifies the HMAC-SHA1-96 authentication algorithm. This mandatory parameter can be only the **sha1** keyword in the current release.

Including the optional **no-encrypt** keyword means that when you display the IPsec configuration, the key is displayed in its unencrypted form and also saved as unencrypted.

The *key* variable must be 40 hexadecimal characters. To change an existing key, you must also specify a different SPI value. You cannot just change the key without also specifying a different SPI, too. For example, in an interface context where you intend to change a key, you must type a different SPI value—which occurs before the key parameter on the command line—before you type the new key. The example in "IPsec for OSPF V3 configuration"illustrates this requirement.

If **no-encrypt** is not entered, then the key will be encrypted. This is the default. The system adds the following in the configuration to indicate that the key is encrypted:

- encrypt = the key string uses proprietary simple crytographic 2-way algorithm.
- encryptb64 = the key string uses proprietary base64 crytographic 2-way algorithm.

This example results in the configuration shown in the screen output that follows. Note that because the optional **no-encrypt** keyword was omitted, the display of the key has the encrypted form by default.

```
interface ethernet 1/1/2
 enable
 ip address 10.3.3.1/8
 ipv6 address 2001:db8:3::1/64
 ipv6 ospf area 1
 ipv6 ospf authentication ipsec spi 429496795 esp sha1 encryptb64
$ITJkQG5HWnw4M09tWVd
```

Configuring IPsec for an area

This application of the **area** command (for IPsec) applies to all of the interfaces that belong to an area unless an interface has its own IPsec configuration. (As described in "Disabling IPsec on an interface" on page 253, the interface IPsec can be operationally disabled if necessary.) To configure IPsec for an area in the IPv6 router OSPF context, proceed as in the following example.

```
Brocade(config-ospf6-router)#area 2 auth ipsec spi 400 esp sha1
abcef12345678901234fedcba098765432109876
```

Syntax: **area** *area-id* **authentication ipsec spi** *spinum* **esp sha1** [**no-encrypt**] *key*

The **no** form of this command deletes IPsec from the area.

The **area** command and the *area-id* variable specify the area for this IPsec configuration. The *area-id* can be an integer in the range 0–2,147,483,647 or have the format of an IP address.

The **authentication** keyword specifies that the function to specify for the area is packet authentication.

The **ipsec** keyword specifies that IPsec is the protocol that authenticates the packets.

The **spi** keyword and the *spinum* variable specify the index that points to the security association. The near-end and far-end values for spinum must be the same. The range for *spinum* is decimal 256–4294967295.

The mandatory **esp** keyword specifies ESP (rather than authentication header) as the protocol to provide packet-level security. In the current release, this parameter can be **esp** only.

The **sha1** keyword specifies the HMAC-SHA1-96 authentication algorithm. This mandatory parameter can be only the **sha1** keyword in the current release.

Including the optional **no-encrypt** keyword means that the 40-character key is not encrypted upon either its entry or its display. The key must be 40 hexadecimal characters.

If **no-encrypt** is not entered, then the key will be encrypted. This is the default. The system adds the following in the configuration to indicate that the key is encrypted:

- encrypt = the key string uses proprietary simple crytographic 2-way algorithm.
- encryptb64 = the key string uses proprietary base64 crytographic 2-way algorithm.

The configuration in the preceding example results in the configuration for area 2 that is illustrated in the following example.

```
 ipv6 router ospf
  area 0
  area 1
  area 2
  area 2 auth ipsec spi 400 esp sha1 abcef12345678901234fedcba098765432109876
```

Configuring IPsec for a virtual link

IPsec on a virtual link has a global configuration.

To configure IPsec on a virtual link, enter the IPv6 router OSPF context of the CLI and proceed as the following example illustrates. (Note the **no-encrypt** option in this example.)

```
Brocade(config-ospf6-router)#area 1 vir 10.2.2.2 auth ipsec spi 360 esp sha1
no-encrypt 1234567890098765432112345678990987654321
```

Syntax: [no] **area** *area-id* **virtual** *nbrid* **authentication ipsec spi** *spinum* **esp sha1** [**no-encrypt**] *key*

The **no** form of this command deletes IPsec from the virtual link.

The **area** command and the *area-id* variable specify the area is to be configured. The *area-id* can be an integer in the range 0–2,147,483,647 or have the format of an IP address.

The **virtual** keyword indicates that this configuration applies to the virtual link identified by the subsequent variable *nbrid*. The variable *nbrid* is in dotted decimal notation of an IP address.

The **authentication** keyword specifies that the function to specify for the area is packet authentication.

The **ipsec** keyword specifies that IPsec is the protocol that authenticates the packets.

The **spi** keyword and the *spinum* variable specify the index that points to the security association. The near-end and far-end values for spinum must be the same. The range for *spinum* is decimal 256–4294967295.

The mandatory **esp** keyword specifies ESP (rather than authentication header) as the protocol to provide packet-level security. In the current release, this parameter can be **esp** only.

The **sha1** keyword specifies the HMAC-SHA1-96 authentication algorithm. This mandatory parameter can be only the **sha1** keyword in the current release.

Including the optional **no-encrypt** keyword means that the 40-character key is not encrypted in **show** command displays. If **no-encrypt** is not entered, then the key will be encrypted. This is the default. The system adds the following in the configuration to indicate that the key is encrypted:

- encrypt = the key string uses proprietary simple crytographic 2-way algorithm.
- encryptb64 = the key string uses proprietary base64 crytographic 2-way algorithm.

This example results in the following configuration.

```
area 1 virtual-link 10.2.2.2
area 1 virtual-link 10.2.2.2 authentication ipsec spi 360 esp sha1 no-encrypt 12
3456789009876543211234567899098765432l
```

Disabling IPsec on an interface

For the purpose of troubleshooting, you can operationally disable IPsec on an interface by using the **ipv6 ospf authentication ipsec disable** command in the CLI context of a specific interface. This command disables IPsec on the interface whether its IPsec configuration is the area's IPsec configuration or is specific to that interface. The output of the **show ipv6 ospf interface command** shows the current setting for the disable command.

To disable IPsec on an interface, go to the CLI context of the interface and proceed as in the following example.

```
Brocade(config-if-e10000-1/1/2)#ipv6 ospf auth ipsec disable
```

Syntax: [no] **ipv6 ospf authentication ipsec disable**

The **no** form of this command restores the area and interface-specific IPsec operation.

Changing the key rollover timer

Configuration changes for authentication takes effect in a controlled manner through the key rollover procedure as specified in RFC 4552, Section 10.1. The key rollover timer controls the timing of the configuration changeover. The key rollover timer can be configured in the IPv6 router OSPF context, as the following example illustrates.

```
Brocade(config-ospf6-router)#key-rollover-interval 200
```

Syntax: **key-rollover-interval** *time*

The range for the key-rollover-interval is 0–14400 seconds. The default is 300 seconds.

Clearing IPsec statistics

This section describes the **clear ipsec statistics** command for clearing statistics related to IPsec. The command resets to 0 the counters (which you can view as a part of IPSecurity Packet Statistics). The counters hold IPsec packet statistics and IPsec error statistics. The following example illustrates the **show ipsec statistics** output.

```
Brocade#show ipsec statistics
                     IPSecurity Statistics
secEspCurrentInboundSAs 1          ipsecEspTotalInboundSAs:  2
secEspCurrentOutboundSA 1          ipsecEspTotalOutboundSAs: 2
                   IPSecurity Packet Statistics
secEspTotalInPkts:      20         ipsecEspTotalInPktsDrop:  0
secEspTotalOutPkts:     84
                   IPSecurity Error Statistics
secAuthenticationErrors 0
secReplayErrors:        0          ipsecPolicyErrors:        13
secOtherReceiveErrors:  0          ipsecSendErrors:          0
secUnknownSpiErrors:    0
```

To clear the statistics, enter the **clear ipsec statistics** command as in the following example.

```
Brocade#clear ipsec statistics
```

Syntax: **clear ipsec statistics**

This command takes no parameters.

Displaying OSPF V3 Information

You can display the information for the following OSPF V3 parameters:

- Areas
- Link state databases
- Interfaces
- Memory usage
- Neighbors
- Redistributed routes
- Routes
- SPF

- Virtual links
- Virtual neighbors
- IPsec

Displaying OSPF V3 area information

To display global OSPF V3 area information for the Brocade device, enter the following command at any CLI level.

```
Brocade#show ipv6 ospf area
Area 0:
  Interface attached to this area: loopback 2 ethe 1/1/3 tunnel 2
  Number of Area scoped LSAs is 6
  Statistics of Area 0:
    SPF algorithm executed 16 times
    SPF last updated: 335256 sec ago
    Current SPF node count: 3
      Router: 2 Network: 1
      Maximum of Hop count to nodes: 2
...
```

Syntax: **show ipv6 ospf area** [*area id*]

You can specify the *area-id* parameter in the following formats:

- As an IPv4 address, for example, 192.168.1.1.
- As a numerical value from 0–2,147,483,647.

The *area-id* parameter restricts the display to the specified OSPF area.

This display shows the following information.

TABLE 43 **OSPF V3 area information fields**

Field	Description
Area	The area number.
Interface attached to this area	The router interfaces attached to the area.
Number of Area scoped LSAs	Number of LSAs with a scope of the specified area.
SPF algorithm executed	The number of times the OSPF Shortest Path First (SPF) algorithm is executed within the area.
SPF last updated	The interval in seconds that the SPF algorithm was last executed within the area.
Current SPF node count	The current number of SPF nodes in the area.
Router	Number of router LSAs in the area.
Network	Number of network LSAs in the area.
Indx	The row number of the entry in the router OSPF area table.
Area	The area number.
Maximum hop count to nodes.	The maximum number of hop counts to an SPF node within the area.

Displaying OSPF V3 database information

You can display a summary of the link state database or detailed information about a specified LSA type.

To display a summary of a device link state database, enter the **show ipv6 ospf database** command at any CLI level.

```
Brocade#show ipv6 ospf database
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum  Len
0               Link 000001e6 192.168.223.223 800000ab 1547 8955   68
0               Link 000000d8 10.1.1.1         800000aa 1295 0639   68
0               Link 00000185 192.168.223.223 800000ab 1481 7e6b   56
0               Iap  00000077 192.168.223.223 800000aa 1404 966a   56
0               Rtr  00000124 192.168.223.223 800000b0 1397 912c   40
0               Net  00000016 192.168.223.223 800000aa 1388 1b09   32
0               Iap  000001d1 192.168.223.223 800000bd 1379 a072   72
0               Iap  000000c3 10.1.1.1         800000ae 1325 e021   52
0               Rtr  00000170 10.1.1.1         800000ad 1280 af8e   40
N/A             Extn 00000062 192.168.223.223 800000ae 1409 0ca7   32
N/A             Extn 0000021d 192.168.223.223 800000a8 1319 441e   32
```

Syntax: **show ipv6 ospf database** [**advrtr** *ipv4-address* | **as-external** | **extensive** | **inter-prefix** | **inter-router** | **intra-prefix** | **link** | **link-id** *number* | **network** | **router** [**scope** *area-id* | **as** | **link**]]

The **advrtr** *ipv4-address* parameter displays detailed information about the LSAs for a specified advertising router only.

The **as-external** keyword displays detailed information about the AS externals LSAs only.

The **extensive** keyword displays detailed information about all LSAs in the database.

The **inter-prefix** keyword displays detailed information about the inter-area prefix LSAs only.

The **inter-router** keyword displays detailed information about the inter-area router LSAs only.

The **intra-prefix** keyword displays detailed information about the intra-area prefix LSAs only.

The **link** keyword displays detailed information about the link LSAs only.

The **link-id** *number* parameter displays detailed information about the specified link LSAs only.

The **network** *number* displays detailed information about the network LSAs only.

The **router** *number* displays detailed information about the router LSAs only.

The **scope** *area-id* parameter displays detailed information about the LSAs for a specified area, AS, or link.

This display shows the following information.

TABLE 44 OSPF V3 database summary fields

Field	Description
Area ID	The OSPF area in which the Brocade device resides.
Type	Type of LSA. LSA types can be the following: • Rtr – Router LSAs (Type 1). • Net – Network LSAs (Type 2). • Inap – Inter-area prefix LSAs for ABRs (Type 3). • Inar – Inter-area router LSAs for ASBRs (Type 4). • Extn – AS external LSAs (Type 5). • Link – Link LSAs (Type 8). • Iap – Intra-area prefix LSAs (Type 9).
LS ID	The ID of the LSA, in hexadecimal, from which the device learned this route.
Adv Rtr	The device that advertised the route.
Seq(Hex)	The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps it with a sequence number to enable the Brocade device and other OSPF routers to determine which LSA for a given route is the most recent.
Age	The age of the LSA, in seconds.
Chksum	A checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The Brocade device uses the checksum to verify that the packet is not corrupted.
Len	The length, in bytes, of the LSA.

For example, to display detailed information about all LSAs in the database, enter the **show ipv6 ospf database extensive** command at any CLI level.

```
Brocade#show ipv6 ospf database extensive
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
0               Link 00000031 10.1.1.1         80000001 35   6db9   56
     Router Priority: 1
     Options: V6E---R--
     LinkLocal Address: 2001:db8::1
     Number of Prefix: 1
     Prefix Options:
     Prefix: 2001:db8::/64
    ...
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
0               Iap  00000159 192.168.223.223 800000ab 357  946b    56
     Number of Prefix: 2
     Referenced LS Type: Network
     Referenced LS ID: 00000159
     Referenced Advertising Router: 192.168.223.223
     Prefix Options:  Metric: 0
     Prefix: 2001:db8::/64
     Prefix Options:  Metric: 0
     Prefix: 2001:db8:46a::/64
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
0               Rtr  00000039 192.168.223.223 800000b1 355  8f2d    40
  Capability Bits: --E-
  Options: V6E---R--
  Type: Transit Metric: 1
  Interface ID: 00000058  Neighbor Interface ID: 00000058
  Neighbor Router ID: 192.168.223.223
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
0               Net  000001f4 192.168.223.223 800000ab 346  190a    32
     Options: V6E---R--
     Attached Router: 192.168.223.223
     Attached Router: 10.1.1.1
...
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
N/A             Extn 000001df 192.168.223 800000af 368  0aa8   32
     Bits: E
     Metric: 00000001
     Prefix Options:
     Referenced LSType: 0
     Prefix: 2001:db8::/16
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
1               Inap 0000011d 10.1.1.188      80000001 124  25de    36
     Metric: 2
     Prefix Options:
     Prefix: 2001:db8:2::/64
Area ID         Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum   Len
0               Inar 0000005b 10.1.1.198      80000001 990  dbad    32
     Options: V6E---R--
     Metric: 1
     Destination Router ID:10.1.1.188
```

NOTE

Portions of this display are truncated for brevity. The purpose of this display is to show all possible fields that might display rather than to show complete output.

The fields that display depend upon the LSA type as shown in the following table.

TABLE 45 OSPF V3 detailed database information fields

Field	Description
Router LSA (Type 1) (Rtr) fields	
Capability Bits	A bit that indicates the capability of the Brocade device. The bit can be set to one of the following: • B – The device is an area border router. • E – The device is an AS boundary router. • V – The device is a virtual link endpoint. • W – The device is a wildcard multicast receiver.
Options	A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following: V6 – The device should be included in IPv6 routing calculations. E – The device floods AS-external-LSAs as described in RFC 2740. MC – The device forwards multicast packets as described in RFC 1586. N – The device handles type 7 LSAs as described in RFC 1584. R – The originator is an active router. DC –The device handles demand circuits.
Type	The type of interface. Possible types can be the following: • Point-to-point – A point-to-point connection to another router. • Transit – A connection to a transit network. • Virtual link – A connection to a virtual link.
Metric	The cost of using this router interface for outbound traffic.
Interface ID	The ID assigned to the router interface.
Neighbor Interface ID	The interface ID that the neighboring router has been advertising in hello packets sent on the attached link.
Neighbor Router ID	The router ID (IPv4 address) of the neighboring router that advertised the route. (By default, the Brocade router ID is the IPv4 address configured on the lowest numbered loopback interface. If the Brocade device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device.)

TABLE 45 OSPF V3 detailed database information fields (Continued)

Field	Description
Network LSA (Type 2) (Net) fields	
Options	A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following: V6 – The device should be included in IPv6 routing calculations. E – The device floods AS-external-LSAs as described in RFC 2740. MC – The device forwards multicast packets as described in RFC 1586. N – The device handles type 7 LSAs as described in RFC 1584. R – The originator is an active router. DC –The device handles demand circuits.
Attached Router	The address of the neighboring router that advertised the route.
Inter-Area Prefix LSA (Type 3) (Inap) fields	
Metric	The cost of the route.
Prefix Options	An 8-bit field describing various capabilities associated with the prefix.
Prefix	The IPv6 prefix included in the LSA.
Inter-Area Router LSA (Type 4) (Inar) fields	
Options	A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following: V6 – The device should be included in IPv6 routing calculations. E – The device floods AS-external-LSAs as described in RFC 2740. MC – The device forwards multicast packets as described in RFC 1586. N – The device handles type 7 LSAs as described in RFC 1584. R – The originator is an active router. DC –The device handles demand circuits.
Metric	The cost of the route.
Destination Router ID	The ID of the router described in the LSA.
AS External LSA (Type 5) (Extn) fields	
Bits	The bit can be set to one of the following: • E – If bit E is set, a Type 2 external metric. If bit E is zero, a Type 1 external metric. • F – A forwarding address is included in the LSA. • T – An external route tag is included in the LSA.
Metric	The cost of this route, which depends on bit E.
Prefix Options	An 8-bit field describing various capabilities associated with the prefix.
Referenced LS Type	If non-zero, an LSA with this LS type is associated with the LSA.
Prefix	The IPv6 prefix included in the LSA.
Link LSA (Type 8) (Link) fields	
Router Priority	The router priority of the interface attaching the originating router to the link.
Options	The set of options bits that the router would like set in the network LSA that will be originated for the link.
Link Local Address	The originating router link-local interface address on the link.
Number of Prefix	The number of IPv6 address prefixes contained in the LSA.

TABLE 45 OSPF V3 detailed database information fields (Continued)

Field	Description
Prefix Options	An 8-bit field of capabilities that serve as input to various routing calculations: • NU – The prefix is excluded from IPv6 unicast calculations. • LA – The prefix is an IPv6 interface address of the advertising router. • MC – The prefix is included in IPv6 multicast routing calculations.
Prefix	The IPv6 prefix included in the LSA.
Intra-Area Prefix LSAs (Type 9) (Iap) fields	
Number of Prefix	The number of prefixes included in the LSA.
Referenced LS Type, Referenced LS ID	Identifies the router-LSA or network-LSA with which the IPv6 address prefixes are associated.
Referenced Advertising Router	The address of the neighboring router that advertised the route.
Prefix Options	An 8-bit field describing various capabilities associated with the prefix.
Metric	The cost of using the advertised prefix.
Prefix	The IPv6 prefix included in the LSA.
Number of Prefix	The number of prefixes included in the LSA.

Displaying OSPF V3 interface information

You can display a summary of information for all OSPF V3 interfaces or detailed information about a specified OSPF V3 interface.

To display a summary of OSPF V3 interfaces, enter the **show ipv6 ospf interface** command at any CLI level.

```
Brocade#show ipv6 ospf interface
Interface     OSPF      Status State     Area
ethe 1/1/1              up
ethe 1/1/2    enabled   up     DR        0
ethe 1/1/3    disabled  down
loopback 2    enabled   up     Loopback  0
tunnel 1      disabled  down
tunnel 2      enabled   up     P2P       0
tunnel 6                up
```

Syntax: **show ipv6 ospf interface** [**ethernet** *port* | **loopback** *number* | **tunnel** *number* | **ve** *number*]

The **ethernet** | **loopback** | **tunnel** | **ve** parameter specifies the interface for which to display information. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a loopback, tunnel, or VE interface, also specify the number associated with the interface.

This display shows the following information.

TABLE 46 Summary of OSPF V3 interface information

Field	Description
Interface	The interface type, and the port number or number of the interface.
OSPF	The state of OSPF V3 on the interface. Possible states include the following: • Enabled. • Disabled.
Status	The status of the link. Possible status include the following: • Up. • Down.
State	The state of the interface. Possible states includes the following: • DR – The interface is functioning as the Designated Router for OSPF V3. • BDR – The interface is functioning as the Backup Designated Router for OSPF V3. • Loopback – The interface is functioning as a loopback interface. • P2P – The interface is functioning as a point-to-point interface. • Passive – The interface is up but it does not take part in forming an adjacency. • Waiting – The interface is trying to determine the identity of the BDR for the network. • None – The interface does not take part in the OSPF interface state machine. • Down – The interface is unusable. No protocol traffic can be sent or received on such a interface. • DR other – The interface is a broadcast or NBMA network on which another router is selected to be the DR.
Area	The OSPF area to which the interface belongs.

For example, to display detailed information about Ethernet interface 2, enter the **show ipv6 ospf interface ethernet** command at any level of the CLI.

```
Brocade#show ipv6 ospf interface ethernet 1/1/3
ethe 1/1/3 is up, type BROADCAST
  IPv6 Address:
      2001:db8:46a::1/64
      2001:db8::106/64
  Instance ID 0, Router ID 192.168.223.223
  Area ID 0, Cost 1
  State DR, Transmit Delay 1 sec, Priority 1
  Timer intervals :
    Hello 10, Dead 40, Retransmit 5
  DR:192.168.223.223 BDR:10.1.1.1  Number of I/F scoped LSAs is 2
  DRElection:     5 times, DelayedLSAck:   523 times
  Neighbor Count = 1,   Adjacent Neighbor Count= 1
     Neighbor:
      10.1.1.1 (BDR)
    Statistics of interface ethe 1/1/3:
      Type       tx   rx tx-byte rx-byte
      Unknown     0    0       0       0
      Hello    3149 3138 1259284 1255352
      DbDesc      7    6     416     288
      LSReq       2    2      80     152
      LSUpdate 1508  530  109508   39036
      LSAck     526 1398   19036   54568
```

This display shows the following information.

TABLE 47 Detailed OSPF V3 interface information

Field	Description
Interface status	The status of the interface. Possible status includes the following: • Up. • Down.
Type	The type of OSPF V3 circuit running on the interface. Possible types include the following: • BROADCAST • POINT TO POINT UNKNOWN
IPv6 Address	The IPv6 address(es) assigned to the interface.
Instance ID	An identifier for an instance of OSPF V3.
Router ID	The IPv4 address of the Brocade device. By default, the Brocade router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device.
Area ID	The IPv4 address or numerical value of the area in which the interface belongs.
Cost	The overhead required to send a packet through the interface.
State	The state of the interface. Possible states include the following: • DR – The interface is functioning as the Designated Router for OSPF V3. • BDR – The interface is functioning as the Backup Designated Router for OSPF V3. • Loopback – The interface is functioning as a loopback interface. • P2P – The interface is functioning as a point-to-point interface. • Passive – The interface is up but it does not take part in forming an adjacency. • Waiting – The interface is trying to determine the identity of the BDR for the network. • None – The interface does not take part in the OSPF interface state machine. • Down – The interface is unusable. No protocol traffic can be sent or received on such a interface. • DR other – The interface is a broadcast or NBMA network on which another router is selected to be the DR.
Transmit delay	The amount of time, in seconds, it takes to transmit Link State Updates packets on the interface.
Priority	The priority used when selecting the DR and the BDR. If the priority is 0, the interface does not participate in the DR and BDR election.
Timer intervals	The interval, in seconds, of the hello-interval, dead-interval, and retransmit-interval timers.
DR	The router ID (IPv4 address) of the DR.
BDR	The router ID (IPv4 address) of the BDR.
Number of I/F scoped LSAs	The number of interface LSAs scoped for a specified area, AS, or link.
DR Election	The number of times the DR election occurred.
Delayed LSA Ack	The number of the times the interface sent a delayed LSA acknowledgement.
Neighbor Count	The number of neighbors to which the interface is connected.
Adjacent Neighbor Count	The number of neighbors with which the interface has formed an active adjacency.

TABLE 47 **Detailed OSPF V3 interface information (Continued)**

Field	Description
Neighbor	The router ID (IPv4 address) of the neighbor. This field also identifies the neighbor as a DR or BDR, if appropriate.
Interface statistics	The following statistics are provided for the interface: • Unknown – The number of Unknown packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Unknown packets. • Hello – The number of Hello packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Hello packets. • DbDesc – The number of Database Description packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Database Description packets. • LSReq – The number of link-state requests transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state requests. • LSUpdate – The number of link-state updates transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state requests. • LSAck – The number of link-state acknowledgements transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state acknowledgements.

Displaying OSPF V3 memory usage

To display information about OSPF V3 memory usage, enter the **show ipv6 ospf memory** command at any level of the CLI.

```
Brocade#show ipv6 ospf memory
Total Static Memory Allocated : 5829 bytes
Total Dynamic Memory Allocated : 0 bytes
Memory Type               Size       Allocated  Max-alloc  Alloc-Fails
MTYPE_OSPF6_TOP           0          0          0          0
MTYPE_OSPF6_LSA_HDR       0          0          0          0
MTYPE_OSPF6_RMAP_COMPILED 0          0          0          0
MTYPE_OSPF6_OTHER         0          0          0          0
MTYPE_THREAD_MASTER       0          0          0          0
MTYPE_OSPF6_AREA          0          0          0          0
MTYPE_OSPF6_AREA_RANGE    0          0          0          0
MTYPE_OSPF6_SUMMARY_ADDRE 0          0          0          0
MTYPE_OSPF6_IF            0          0          0          0
MTYPE_OSPF6_NEIGHBOR      0          0          0          0
MTYPE_OSPF6_ROUTE_NODE    0          0          0          0
MTYPE_OSPF6_ROUTE_INFO    0          0          0          0
MTYPE_OSPF6_PREFIX        0          0          0          0
MTYPE_OSPF6_LSA           0          0          0          0
MTYPE_OSPF6_VERTEX        0          0          0          0
MTYPE_OSPF6_SPFTREE       0          0          0          0
MTYPE_OSPF6_NEXTHOP       0          0          0          0
MTYPE_OSPF6_EXTERNAL_INFO 0          0          0          0
MTYPE_THREAD              0          0          0          0
```

Syntax: **show ipv6 ospf memory**

This display shows the following information.

TABLE 48 OSPF V3 memory usage information

Field	Description
Total Static Memory Allocated	A summary of the amount of static memory allocated, in bytes, to OSPF V3.
Total Dynamic Memory Allocated	A summary of the amount of dynamic memory allocated, in bytes, to OSPF V3.
Memory Type	The type of memory used by OSPF V3. (This information is for use by Brocade technical support in case of a problem.)
Size	The size of a memory type.
Allocated	The amount of memory currently allocated to a memory type.
Max-alloc	The maximum amount of memory that was allocated to a memory type.
Alloc-Fails	The number of times an attempt to allocate memory to a memory type failed.

Displaying OSPF V3 neighbor information

You can display a summary of OSPF V3 neighbor information for the Brocade device or detailed information about a specified neighbor.

To display a summary of OSPF V3 neighbor information for the device, enter the **show ipv6 ospf neighbor** command at any CLI level.

```
Brocade#show ipv6 ospf neighbor
RouterID        Pri State    DR              BDR             Interface[State]
10.1.1.1           1 Full    192.168.223.223 10.1.1.1         ethe 1/1/3  [DR]
```

Syntax: **show ipv6 ospf neighbor [router-id** *ipv4-address*]

The **router-id** *ipv4-address* parameter displays only the neighbor entries for the specified router.

This display shows the following information.

TABLE 49 Summary of OSPF V3 neighbor information

Field	Description
Router ID	The IPv4 address of the neighbor. By default, the Brocade router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device.
Pri	The OSPF V3 priority of the neighbor. The priority is used during election of the DR and BDR.
State	The state between the Brocade device and the neighbor. The state can be one of the following: • Down • Attempt • Init • 2-Way • ExStart • Exchange • Loading • Full
DR	The router ID (IPv4 address) of the DR.

TABLE 49 Summary of OSPF V3 neighbor information (Continued)

Field	Description
BDR	The router ID (IPv4 address) of the BDR.
Interface [State]	The interface through which the router is connected to the neighbor. The state of the interface can be one of the following: • DR – The interface is functioning as the Designated Router for OSPF V3. • BDR – The interface is functioning as the Backup Designated Router for OSPF V3. • Loopback – The interface is functioning as a loopback interface. • P2P – The interface is functioning as a point-to-point interface. • Passive – The interface is up but it does not take part in forming an adjacency. • Waiting – The interface is trying to determine the identity of the BDR for the network. • None – The interface does not take part in the OSPF interface state machine. • Down – The interface is unusable. No protocol traffic can be sent or received on such a interface. • DR other – The interface is a broadcast or NBMA network on which another router is selected to be the DR.

For example, to display detailed information about a neighbor with the router ID of 1.1.1.1, enter the **show ipv6 ospf neighbor router-id** command at any CLI level.

```
Brocade#show ipv6 ospf neighbor router-id 3.3.3.3
RouterID        Pri State    DR              BDR             Interface[State]
10.3.3.3           1 Full    10.3.3.3        10.1.1.1         ve 10    [BDR]
DbDesc bit for this neighbor: --s
Nbr Ifindex of this router: 1
Nbr DRDecision: DR 10.3.3.3, BDR 10.1.1.1
Last received DbDesc: opt:xxx ifmtu:0 bit:--s seqnum:0
Number of LSAs in DbDesc retransmitting: 0
Number of LSAs in SummaryList: 0
Number of LSAs in RequestList: 0
Number of LSAs in RetransList: 0
SeqnumMismatch 0 times, BadLSReq 0 times
OnewayReceived 0 times, InactivityTimer 0 times
DbDescRetrans 0 times, LSReqRetrans 0 times
LSUpdateRetrans 1 times
LSAReceived 12 times, LSUpdateReceived 6 times
```

This display shows the following information.

TABLE 50 Detailed OSPF V3 neighbor information

Field	Description
Router ID	For information about this field, refer to Table 49 on page 265.
Pri	For information about this field, refer to Table 49 on page 265.
State	For information about this field, refer to Table 49 on page 265.
DR	For information about this field, refer to Table 49 on page 265.
BDR	For information about this field, refer to Table 49 on page 265.
Interface [State]	For information about this field, refer to Table 49 on page 265.

TABLE 50 Detailed OSPF V3 neighbor information (Continued)

Field	Description
DbDesc bit...	The Database Description packet, which includes 3 bits of information: • The first bit can be “i” or “-”. “i” indicates the inet bit is set. “-” indicates the inet bit is not set. • The second bit can be “m” or “-”. “m” indicates the more bit is set. “-” indicates the more bit is not set. • The third bit can be “m” or “s”. An “m” indicates the master. An “s” indicates standby.
Index	The ID of the LSA from which the neighbor learned of the router.
DR Decision	The router ID (IPv4 address) of the neighbor elected DR and BDR.
Last Received Db Desc	The content of the last database description received from the specified neighbor.
Number of LSAs in Db Desc retransmitting	The number of LSAs that need to be retransmitted to the specified neighbor.
Number of LSAs in Summary List	The number of LSAs in the neighbor summary list.
Number of LSAs in Request List	The number of LSAs in the neighbor request list.
Number of LSAs in Retransmit List	The number of LSAs in the neighbor retransmit list.
Seqnum Mismatch	The number of times sequence number mismatches occurred.
BadLSReq	The number of times the neighbor received a bad link-state request from the Brocade device.
One way received	The number of times a hello packet, which does not mention the router, is received from the neighbor. This omission in the hello packet indicates that the communication with the neighbor is not bidirectional.
Inactivity Timer	The number of times that the neighbor inactivity timer expired.
Db Desc Retransmission	The number of times sequence number mismatches occurred.
LSReqRetrans	The number of times the neighbor retransmitted link-state requests to the Brocade device.
LSUpdateRetrans	The number of times the neighbor retransmitted link-state updates to the Brocade device.
LSA Received	The number of times the neighbor received LSAs from the Brocade device.
LS Update Received	The number of times the neighbor received link-state updates from the Brocade device.

Displaying routes redistributed into OSPF V3

You can display all IPv6 routes or a specified IPv6 route that the Brocade device has redistributed into OSPF V3.

To display all IPv6 routes that the device has redistributed into OSPF V3, enter the **show ipv6 ospf redistribute route** command at any level of the CLI.

```
Brocade#show ipv6 ospf redistribute route

Id     Prefix                                     Protocol  Metric Type  Metric
 snIpAsPathAccessListStringRegExpression
1      2001:db8::/16                              Static    Type-2       1
2      2001:db8::/32                              Static    Type-2       1
```

Syntax: **show ipv6 ospf redistribute route** [*ipv6-prefix*]

The *ipv6-prefix* parameter specifies an IPv6 network prefix. (You do not need to specify the length of the prefix.)

For example, to display redistribution information for the prefix 2002::, enter the **show ipv6 ospf redistribute route** command at any level of the CLI.

```
Brocade#show ipv6 ospf redistribute route 2001:db8::
Id     Prefix               Protocol  Metric Type  Metric
1      2001:db8::/16        Static    Type-2       1
```

These displays show the following information.

TABLE 51 **OSPF V3 redistribution information**

Field	Description
ID	An ID for the redistributed route.
Prefix	The IPv6 routes redistributed into OSPF V3.
Protocol	The protocol from which the route is redistributed into OSPF V3. Redistributed protocols can be the following: • RIP – RIPng. • Static – IPv6 static route table. • Connected – A directly connected network.
Metric Type	The metric type used for routes redistributed into OSPF V3. The metric type can be the following: • Type-1 – Specifies a small metric (2 bytes). • Type-2 – Specifies a big metric (3 bytes).
Metric	The value of the default redistribution metric, which is the OSPF cost of redistributing the route into OSPF V3.

Displaying OSPF V3 route information

You can display the entire OSPF V3 route table for the Brocade device or only the route entries for a specified destination.

To display the entire OSPF V3 route table for the device, enter the **show ipv6 ospf routes** command at any level of the CLI.

```
  Brocade#show ipv6 ospf routes
Current Route count: 4
    Intra: 4 Inter: 0 External: 0 (Type1 0/Type2 0)
    Equal-cost multi-path: 0
    Destination                    Options   Area             Cost Type2 Cost
    Next Hop Router                Outgoing Interface
*IA 2001db8::/64                   V6E---R-- 0.0.0.0             1   0
    ::                             ethe 1/1/2
*IA 2001db8:46a::/64               V6E---R-- 0.0.0.0             1   0
    ::                             ethe 1/1/2
*IA 2001db8::1/128                 --------- 0.0.0.0             0   0
    ::                             loopback 2
*IA 2001db8::2/128                 V6E---R-- 0.0.0.0             1   0
    2001db8:2e0:52ff:fe91:bb37     ethe 1/1/2
```

Syntax: **show ipv6 ospf routes** [*ipv6-prefix*]

The *ipv6-prefix* parameter specifies a destination IPv6 prefix. (You do not need to specify the length of the prefix.) If you use this parameter, only the route entries for this destination are shown.

For example, to display route information for the destination prefix 2000:4::, enter the **show ipv6 ospf routes** command at any level of the CLI.

```
 Brocade#show ipv6 ospf routes 2001:db8::
Destination                    Options   Area             Cost Type2 Cost
    Next Hop Router             Outgoing Interface
*IA 2001:db8::/64               V6E---R-- 0.0.0.0             1   0
    ::                          ethe 1/1/2
```

These displays show the following information.

TABLE 52 OSPF V3 route information

Field	Description
Current Route Count (Displays with the entire OSPF V3 route table only)	The number of route entries currently in the OSPF V3 route table.
Intra/Inter/External (Type1/Type2) (Displays with the entire OSPF V3 route table only)	The breakdown of the current route entries into the following route types: • Inter – The number of routes that pass into another area. • Intra – The number of routes that are within the local area. • External1 – The number of type 1 external routes. • External2 – The number of type 2 external routes.
Equal-cost multi-path (Displays with the entire OSPF V3 route table only)	The number of equal-cost routes to the same destination in the OSPF V3 route table. If load sharing is enabled, the router equally distributes traffic among the routes.
Destination	The IPv6 prefixes of destination networks to which the Brocade device can forward IPv6 packets. "*IA" indicates the next router is an intra-area router.

TABLE 52 **OSPF V3 route information (Continued)**

Field	Description
Options	A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following: V6 – The device should be included in IPv6 routing calculations. E – The device floods AS-external-LSAs as described in RFC 2740. MC – The device forwards multicast packets as described in RFC 1586. N – The device handles type 7 LSAs as described in RFC 1584. R – The originator is an active router. DC –The device handles demand circuits.
Area	The area whose link state information has led to the routing table entry's collection of paths.
Cost	The type 1 cost of this route.
Type2 Cost	The type 2 cost of this route.
Next-Hop Router	The IPv6 address of the next router a packet must traverse to reach a destination.
Outgoing Interface	The router interface through which a packet must traverse to reach the next-hop router.

Displaying OSPF V3 SPF information

You can display the following OSPF V3 SPF information:

- SPF node information for a specified area.
- SPF table for a specified area.
- SPF tree for a specified area.

For example, to display information about SPF nodes in area 0, enter the **show ipv6 ospf spf node area** command at any level of the CLI.

```
Brocade#show ipv6 ospf spf node area 0
SPF node for Area 0
SPF node 192.168.223.223,  cost: 0,  hops: 0
 nexthops to node:
 parent nodes:
 child nodes: 192.168.223.223:88

SPF node 192.168.223.223:88,  cost: 1,  hops: 1
 nexthops to node:    :: ethe 1/1/2
 parent nodes: 192.168.223.223
 child nodes: 10.1.1.1:0

SPF node 10.1.1.1:0,  cost: 1,  hops: 2
 nexthops to node:    2001:db8:2e0:52ff:fe91:bb37 ethe 1/1/2
 parent nodes: 192.168.223.223:88
 child nodes:
```

Syntax: **show ipv6 ospf spf node area** [*area-id*]

The **node** keyword displays SPF node information.

The **area** *area-id* parameter specifies a particular area. You can specify the *area-id* in the following formats:

- As an IPv4 address; for example, 192.168.1.1.
- As a numerical value from 0– 2,147,483,647.

This display shows the following information.

TABLE 53 OSPF V3 SPF node information

Field	Description
SPF node	Each SPF node is identified by its router ID (IPv4 address). If the node is a child node, it is additionally identified by an interface on which the node can be reached appended to the router ID in the format *router-id*:*interface-id*.
Cost	The cost of traversing the SPF node to reach the destination.
Hops	The number of hops needed to reach the parent SPF node.
Next Hops to Node	The IPv6 address of the next hop-router or the router interface, or both, through which to access the next-hop router.
Parent Nodes	The SPF node parent nodes. A ***parent node*** is an SPF node at the highest level of the SPF tree, which is identified by its router ID.
Child Nodes	The SPF node child nodes. A ***child node*** is an SPF node at a lower level of the SPF tree, which is identified by its router ID and interface on which the node can be reached.

For example, to display the SPF table for area 0, enter the **show ipv6 ospf spf table area** command at any level of the CLI.

```
Brocade#show ipv6 ospf spf table area 0
  SPF table for Area 0
  Destination         Bits Options  Cost  Nexthop                   Interface
R 10.1.1.1            ---- V6E---R-    1  2001:db8:2e0:52ff:fe91:bb37  ethe 1/1/2
N 192.168.223.223[88] ---- V6E---R-    1  ::                        ethe 1/1/2
```

Syntax: **show ipv6 ospf spf table area** *area-id*

The **table** parameter displays the SPF table.

The **area** *area-id* parameter specifies a particular area. You can specify the *area-id* in the following formats:

- As an IPv4 address, for example, 192.168.1.1.
- As a numerical value from 0–2,147,483,647.

This display shows the following information.

TABLE 54 OSPF V3 SPF table

Field	Description
Destination	The destination of a route, which is identified by the following: • "R", which indicates the destination is a router. "N", which indicates the destination is a network. • An SPF node router ID (IPv4 address). If the node is a child node, it is additionally identified by an interface on which the node can be reached appended to the router ID in the format *router-id*:*interface-id*.
Bits	A bit that indicates the capability of the Brocade device. The bit can be set to one of the following: • B – The device is an area border router. • E – The device is an AS boundary router. • V – The device is a virtual link endpoint. • W – The device is a wildcard multicast receiver.
Options	A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following: V6 – The router should be included in IPv6 routing calculations. E – The router floods AS-external-LSAs as described in RFC 2740. MC – The router forwards multicast packets as described in RFC 1586. N – The router handles type 7 LSAs as described in RFC 1584. R – The originator is an active router. DC –The router handles demand circuits.
Cost	The cost of traversing the SPF node to reach the destination.
Next hop	The IPv6 address of the next hop-router.
Interface	The router interface through which to access the next-hop router.

For example, to display the SPF tree for area 0, enter the **show ipv6 ospf spf tree area** command at any level of the CLI.

```
Brocade#show ipv6 ospf spf tree area 0
   SPF tree for Area 0
 +- 192.168.223.223 cost 0
     +- 192.168.223.223:88 cost 1
         +- 10.1.1.1:0 cost 1
```

Syntax: **show ipv6 ospf spf tree area** *area-id*

The **tree** keyword displays the SPF table.

The **area** *area-id* parameter specifies a particular area. You can specify the *area-id* in the following formats:

- As an IPv4 address; for example, 192.168.1.1.
- As a numerical value from 0 – 2,147,483,647.

In this sample output, consider the SPF node with the router ID 192.168.223.223 to be the top (root) of the tree and the local router. Consider all other layers of the tree (192.168.223.223:88 and 10.1.1.1:0) to be destinations in the network. Therefore, traffic destined from router 192.168.223.223 to router 10.1.1.1:0 must first traverse router 192.168.223.223:88.

Displaying IPv6 OSPF virtual link information

To display OSPF V3 virtual link information for the Brocade device, enter the **show ipv6 ospf virtual-link** command at any level of the CLI.

```
Brocade#show ipv6 ospf virtual-link
Index Transit Area ID  Router ID       Interface Address             State
1     1                10.1.1.1         2001:db8::2                   P2P
```

Syntax: **show ipv6 ospf virtual-link**

This display shows the following information.

TABLE 55 **OSPF V3 virtual link information**

Field	Description
Index	An index number associated with the virtual link.
Transit Area ID	The ID of the shared area of two ABRs that serves as a connection point between the two routers.
Router ID	IPv4 address of the router at the other end of the virtual link (virtual neighbor).
Interface Address	The local address used to communicate with the virtual neighbor.
State	The state of the virtual link. Possible states include the following: • P2P – The link is functioning as a point-to-point interface. • DOWN – The link is down.

Displaying OSPF V3 virtual neighbor information

To display OSPF V3 virtual neighbor information for the Brocade device, enter the **show ipv6 ospf virtual-neighbor** command at any level of the CLI.

```
Brocade#show ipv6 ospf virtual-neighbor
Index Router ID        Address                      State     Interface
1     10.1.1.1         2001:db8::1                  Full      ethe 1/1/3
```

Syntax: **show ipv6 ospf virtual-neighbor**

This display shows the following information.

TABLE 56 **OSPF V3 virtual neighbor information**

Field	Description
Index	An index number associated with the virtual neighbor.
Router ID	IPv4 address of the virtual neighbor.
Address	The IPv6 address to be used for communication with the virtual neighbor.

TABLE 56 OSPF V3 virtual neighbor information (Continued)

Field	Description
State	The state between the Brocade device and the virtual neighbor. The state can be one of the following: • Down • Attempt • Init • 2-Way • ExStart • Exchange • Loading • Full
Interface	The IPv6 address of the virtual neighbor.

IPsec examples

This section contains examples of IPsec configuration and the output from the IPsec-specific **show** commands. In addition, IPsec-related information appears in general **show** command output for interfaces and areas.

The **show** commands that are specific to IPsec are:

- **show ipsec sa**
- **show ipsec policy**
- **show ipsec statistics**

The other **show** commands with IPsec-related information are:

- **show ipv6 ospf area**
- **show ipv6 ospf interface**

Showing IPsec security association information

The **show ipsec sa** command displays the IPsec security association databases, as follows.

```
Brocade#show ipsec sa
IPSEC Security Association Database(Entries:8)
SPDID(if)   Dir Encap SPI        Destination         AuthAlg  EncryptAlg
ALL         in  ESP   512        2001:db8:1::1       sha1     Null
eth1/1/1    out ESP   302        ::                  sha1     Null
eth1/1/1    in  ESP   302        2001:db8::          sha1     Null
eth1/1/1    out ESP   512        2001:db8:1::2       sha1     Null
ALL         in  ESP   512        2001:db8:1::1       sha1     Null
eth1/1/2    out ESP   302        ::                  sha1     Null
eth1/1/2    in  ESP   302        2001:db8::          sha1     Null
eth1/1/2    out ESP   512        2001:db8:1::2       sha1     Null
```

Syntax: **show ipsec sa**

Showing IPsec policy

The **show ipsec policy** command displays the database for the IPsec security policies. The fields for this **show** command output appear in the screen output example that follows. However, you should understand the layout and column headings for the display before trying to interpret the information in the example screen.

Each policy entry consists of two categories of information:

- The policy information
- The SA used by the policy

The policy information line in the screen begins with the heading Ptype and also has the headings Dir, Proto, Source (Prefix:TCP.UDP Port), and Destination (Prefix:TCP/UDPPort). The SA line contains the SPDID, direction, encapsulation (always ESP in the current release), the user-specified SPI, For readability, the policy information is described in Table 57, and SA-specific information is in Table 58.

```
Brocade#show ipsec policy
             IPSEC Security Policy Database(Entries:8)
PType  Dir Proto Source(Prefix:TCP/UDP Port)    Destination(Prefix:TCP/UDPPort)
SA: SPDID(if) Dir Encap SPI       Destination
use    in  OSPF  2001:db8::/10:any                 ::/0:any
SA: eth1/1/2  in  ESP   302        FE80::
use    out OSPF  2001:db8::/10:any                 ::/0:any
SA: eth1/1/2  out ESP   302        ::
use    in  OSPF  2001:db8::/10:any                 ::/0:any
SA: eth1/1/1  in  ESP   302        FE80::
use    out OSPF  2001:db8::/10:any                 ::/0:any
SA: eth1/1/1  out ESP   302        ::
use    in  OSPF  2001:db8:1::1/128:any             2001:db8:1::2/128:any
SA: ethALL    in  ESP   512        10:1:1::2
use    out OSPF  2001:db8:1::2/128:any             2001:db8:1::1/128:any
SA: eth1/1/1  out ESP   512        35:1:1::1
use    in  OSPF  2001:db8:1::1/128:any             2001:db8:1::2/128:any
SA: ethALL    in  ESP   512        10:1:1::2
use    out OSPF  2001:db8:1::2/128:any             2001:db8:1::1/128:any
SA: 2:e1/1/2  out ESP   512        2001:db8:1::1
```

Syntax: **show ipsec policy**

This command takes no parameters.

TABLE 57 IPsec policy information

Field	Description
PType	This field contains the policy type. Of the existing policy types, only the “use” policy type is supported, so each entry can have only “use.”
Dir	The direction of traffic flow to which the IPsec policy is applied. Each direction has its own entry.
Proto	The only possible routing protocol for the security policy in the current release is OSPFv3.

TABLE 57 IPsec policy information (Continued)

Field	Description
Source	The source address consists of the IPv6 prefix and the TCP or UDP port identifier.
Destination	The destination address consists of the IPv6 prefix. Certain logical elements have a bearing on the meaning of the destination address and its format, as follows: For IPsec on an interface or area, the destination address is shown as a prefix of 0xFE80 (link local). The solitary “::” (no prefix) indicates a “do not-care” situation because the connection is multicast. In this case, the security policy is enforced without regard for the destination address. For a virtual link (SPDID = 0), the address is required.

TABLE 58 SA used by the policy

Field	Description
SA	This heading points at the SA-related headings for information used by the security policy. Thereafter, on each line of this part of the IPsec entry (which alternates with lines of policy information Table 57), “SA:” points at the fields under those SA-related headings. The remainder of this table describes each of the SA-related items.
SPDID	The Security policy database identifier (SPDID) consists of interface type and Interface ID.
Dir	The Dir field is either ‘in” for inbound or “out” for outbound.
Encap	The type of encapsulation in the current release is ESP.
SPI	Security parameter index.
Destination	The IPv6 address of the destination endpoint. From the standpoint of the near interface and the area, the destination is not relevant and therefore appears as ::/0:any. For a virtual link, both the inbound and outbound destination addresses are relevant.

Showing IPsec statistics

The **show ipsec statistics** command displays the error and other counters for IPsec, as this example shows.

```
Brocade#show ipsec statistics
                       IPSecurity Statistics
secEspCurrentInboundSAs 1          ipsecEspTotalInboundSAs:  2
secEspCurrentOutboundSA 1          ipsecEspTotalOutboundSAs: 2
                   IPSecurity Packet Statistics
secEspTotalInPkts:      19         ipsecEspTotalInPktsDrop:  0
secEspTotalOutPkts:     83
                   IPSecurity Error Statistics
secAuthenticationErrors 0
secReplayErrors:        0          ipsecPolicyErrors:        13
secOtherReceiveErrors:  0          ipsecSendErrors:          0
secAuthenticationErrors 0
secReplayErrors:        0          ipsecPolicyErrors:        13
secOtherReceiveErrors:  0          ipsecSendErrors:          0
secUnknownSpiErrors:    0
```

Syntax: **show ipsec statistics**

This command takes no parameters.

Displaying IPsec configuration for an area

The **show ipv6 ospf area** [*area-id*] command includes information about IPsec for one area or all areas. In the example that follows, the IPsec information is in bold. IPsec is enabled in the first area (area 0) in this example but not in area 3. Note that in area 3, the IPsec key was specified as not encrypted.

```
Brocade(config-ospf6-router)#show ipv6 ospf area
  Authentication: Configured
   KeyRolloverTime(sec): Configured: 25 Current: 20
   KeyRolloverState: Active,Phase1
   Current: None
   New: SPI:400, ESP, SHA1
      Key:$Z|83OmYW{QZ|83OmYW{QZ|83OmYW{QZ|83OmYW{Q
  Interface attached to this area: eth 1/1/1
  Number of Area scoped LSAs is 6
  Sum of Area LSAs Checksum is 0004f7de
  Statistics of Area 0:
    SPF algorithm executed 6 times
    SPF last updated: 482 sec ago
    Current SPF node count: 1
      Router: 1 Network: 0
      Maximum of Hop count to nodes: 0
Area 3:
  Authentication: Not Configured
  Interface attached to this area:
  Number of Area scoped LSAs is 3
```

Syntax: **show ipv6 ospf area** [*area-id*]

The *area-id* parameter restricts the display to the specified OSPF area. You can specify the *area-id* parameter in the following formats:

- An IPv4 address, for example, 192.168.1.1
- A numerical value in the range 0–2,147,483,647

TABLE 59 Area configuration of IPsec

Field	Description
Authentication	This field shows whether or not authentication is configured. If this field says “Not Configured,” the IPsec-related fields (bold in example screen output) are not displayed at all.
KeyRolloverTime	The number of seconds between each initiation of a key rollover. This field shows the configured and current times.
KeyRolloverState	Can be: Not active: key rollover is not active> Active phase 1: rollover is in its first interval. Active phase 2: rollover is in its second interval.
Current	Shows current SPI, authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the current key.

TABLE 59 Area configuration of IPsec (Continued)

Field	Description
New	Shows new SPI (if changed), authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the new key.
Old	Shows old SPI (if changed), authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the old key.

Displaying IPsec for an interface

To see IPsec configuration for a particular interface or all interfaces, use the **show ipv6 ospf interface** command as in the following example (IPsec information in bold).

```
Brocade#show ipv6 ospf interface
eth 1/1/3 is down, type BROADCAST
   Interface is disabled

eth 1/1/8 is up, type BROADCAST
  IPv6 Address:
      2001:db8:18:18:18::1/64
      2001:db8:18:18::/64
  Instance ID 255, Router ID 10.1.1.1
  Area ID 1, Cost 1
  State BDR, Transmit Delay 1 sec, Priority 1
  Timer intervals :
    Hello 10, Hello Jitter 10  Dead 40, Retransmit 5
  Authentication: Enabled
   KeyRolloverTime(sec): Configured: 30 Current: 0
   KeyRolloverState: NotActive
   Outbound: SPI:121212, ESP, SHA1
      Key:1234567890123456789012345678901234567890
   Inbound: SPI:121212, ESP, SHA1
      Key:1234567890123456789012345678901234567890
  DR:10.2.2.2 BDR:10.1.1.1  Number of I/F scoped LSAs is 2
  DRElection:     1 times, DelayedLSAck:    83 times
  Neighbor Count = 1,   Adjacent Neighbor Count= 1
     Neighbor:
      10.2.2.2 (DR)
    Statistics of interface eth 1/1/8:
      Type     tx         rx         tx-byte    rx-byte
      Unknown  0          0          0          0
      Hello    1415       1408       56592      56320
      DbDesc   3          3          804        804
      LSReq    1          1          28         28
      LSUpdate 193        121        15616      9720
      LSAck    85         109        4840       4924
      OSPF messages dropped,no authentication: 0
```

Syntax: **show ipv6 ospf interface** [**ethernet** *slot/port* | **pos** *slot/port* | **loopback** *number* | **tunnel** *number* | **ve** *number*]

TABLE 60 **Area configuration of IPsec**

Field	Description
Authentication	This field shows whether or not authentication is configured. If this field says “Not Configured,” the IPsec-related fields (bold in example screen output) are not displayed at all.
KeyRolloverTime	The number of seconds between each initiation of a key rollover. This field shows the configured and current times.
KeyRolloverState	Can be: Not active: key rollover is not active> Active phase 1: rollover is in its first interval. Active phase 2: rollover is in its second interval.
Current	Shows current SPI, authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the current key.
New (Inbound or Outbound)	Shows new SPI (if changed), authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the new key.
Old (Inbound or Outbound)	Shows old SPI (if changed), authentication algorithm (currently ESP only), encryption algorithm (currently SHA1 only), and the old key.
OSPF messages dropped	Shows the number of packets dropped because the packets failed authentication (for any reason).

Displaying IPsec for a virtual link

To display IPsec for a virtual link, run the **show ipv6 ospf virtual-link brief** or **show ipv6 ospf virtual-link** command, as the following examples illustrate.

```
Brocade#show ipv6 ospf virtual-link brief
Index Transit Area ID  Router ID        Interface Address            State
1     1                10.14.14.14      2001:db8:1:1::1                P2P
```

```
Brocade#show ipv6 ospf virtual-link
Transit Area ID  Router ID        Interface Address            State
1                10.14.14.14      2001:db8:1:1::1                P2P
  Timer intervals(sec) :
    Hello 10, Hello Jitter 10,  Dead 40, Retransmit 5, TransmitDelay 1
  DelayedLSAck:     5 times
  Authentication: Configured
   KeyRolloverTime(sec): Configured: 10 Current: 0
   KeyRolloverState: NotActive
   Outbound: SPI:100004, ESP, SHA1
      Key:1234567890123456789012345678901234567890
   Inbound: SPI:100004, ESP, SHA1
      Key:1234567890123456789012345678901234567890
  Statistics:
      Type     tx         rx         tx-byte    rx-byte
      Unknown  0          0          0          0
      Hello    65         65         2600       2596
      DbDesc   4          4          2752       2992
      LSReq    1          1          232        64
      LSUpdate 11         5          1040       1112
      LSAck    5          8          560        448
      OSPF messages dropped,no authentication: 0
Neighbor: State: Full Address: 2001:db8:44:44::4 Interface: eth 1/1/2
```

Syntax: **show ipv6 ospf virtual-link [brief]**

The optional [brief] keyword limits the display to the Transit, Area ID, Router ID, Interface Address, and State fields for each link.

Changing a key

In this example, the key is changed as illustrated in the two command lines that follow. Note that the SPI value is changed from 300 to 310 to comply with the requirement that you change the SPI when you change the key.

Initial configuration command.

```
Brocade(config-if-e10000-1/1/3)#ipv6 ospf auth ipsec spi 300 esp sha1
no-encrypt 12345678900987655431234567890aabbccddef
```

Command line for changing the key.

```
Brocade(config-if-e10000-1/1/3)#ipv6 ospf auth ipsec spi 310 esp sha1
no-encrypt 989898989009876554321234567890aabbccddef
```

Chapter

BGP (IPv4)

7

Table 61 lists the Border Gateway Protocol (BGP4) features Brocade ICX 6650 devices support. BGP4 features are supported on Brocade ICX 6650 devices running the full Layer 3 software image.

TABLE 61 **Supported BGP4 features**

Feature	Brocade ICX 6650
BGP4	Yes
BGP4 graceful restart	Yes
BGP4 peer group	Yes
Route redistribution	Yes
Route aggregation	Yes
BGP null0 routing	Yes
Route reflection	Yes
BGP filters	Yes
Cooperative BGP4 route filtering	Yes
Route flap dampening	Yes
Multipath load sharing	Yes
Traps for BGP4	Yes

This chapter provides details on how to configure Border Gateway Protocol version 4 (BGP4) on Brocade products using the CLI.

BGP4 is described in RFC 1771. The Brocade implementation fully complies with RFC 1771. The Brocade BGP4 implementation also supports the following RFCs:

- RFC 1745 (OSPF Interactions)
- RFC 1997 (BGP Communities Attributes)
- RFC 2385 (TCP MD5 Signature Option)
- RFC 2439 (Route Flap Dampening)
- RFC 2796 (Route Reflection)
- RFC 2842 (Capability Advertisement)
- RFC 3065 (BGP4 Confederations)

To display BGP4 configuration information and statistics, refer to “Displaying BGP4 information” on page 361.

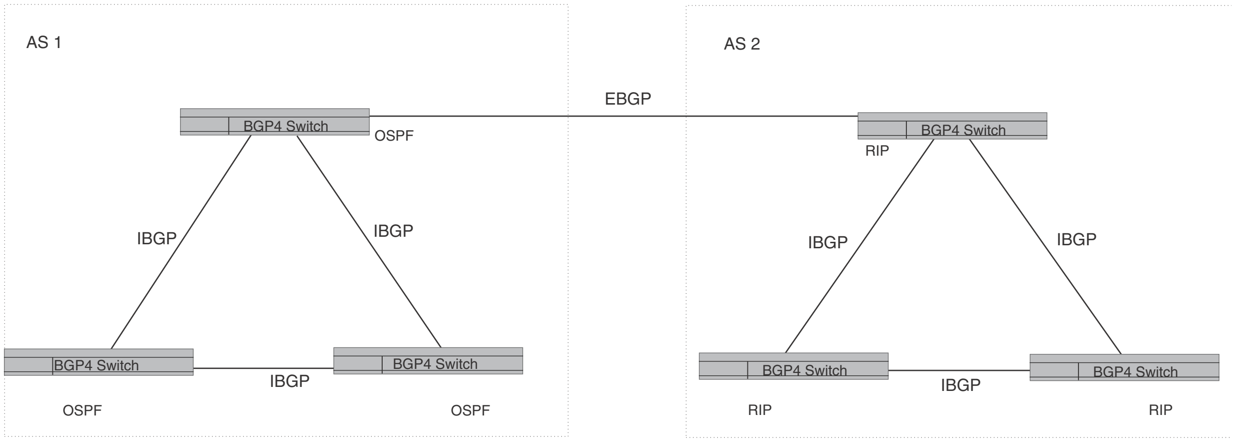
BGP4 overview

Border Gateway Protocol 4 (BGP4) is the standard Exterior Gateway Protocol (EGP) used on the Internet to route traffic between Autonomous Systems (AS) and to maintain loop-free routing. An autonomous system is a collection of networks that share the same routing and administration characteristics. For example, a corporate intranet consisting of several networks under common administrative control might be considered an AS. The networks in an AS can but do not need to run the same routing protocol to be in the same AS, nor do they need to be geographically close.

Routers within an AS can use different Interior Gateway Protocols (IGPs) such as RIP and OSPF to communicate with one another. However, for routers in different autonomous systems to communicate, they need to use an EGP. BGP4 is the standard EGP used by Internet routers and therefore is the EGP implemented on Brocade Layer 3 switches.

Figure 25 on page 282 shows a simple example of two BGP4 autonomous systems. Each AS contains three BGP4 switches. All of the BGP4 switches within an AS communicate using IBGP. BGP4 switches communicate with other autonomous systems using EBGP. Notice that each of the switches also is running an Interior Gateway Protocol (IGP). The switches in AS1 are running OSPF and the switches in AS2 are running RIP. Brocade Layer 3 switches can be configured to redistribute routes among BGP4, RIP, and OSPF. They also can redistribute static routes.

FIGURE 25 Example BGP4 autonomous systems



Relationship between the BGP4 route table and the IP route table

The Brocade Layer 3 switch BGP4 route table can have multiple routes to the same destination, which are learned from different BGP4 neighbors. A BGP4 neighbor is another switch that also is running BGP4. BGP4 neighbors communicate using Transmission Control Protocol (TCP) port 179 for BGP communication. When you configure the Brocade Layer 3 switch for BGP4, one of the configuration tasks you perform is to identify the Layer 3 switch BGP4 neighbors.

Although a Layer 3 Switch BGP4 route table can have multiple routes to the same destination, the BGP4 protocol evaluates the routes and chooses only one of the routes to send to the IP route table. The route that BGP4 chooses and sends to the IP route table is the preferred route and will be used by the Brocade Layer 3 switch. If the preferred route goes down, BGP4 updates the route information in the IP route table with a new BGP4 preferred route.

NOTE
If IP load sharing is enabled and you enable multiple equal-cost paths for BGP4, BGP4 can select more than one equal-cost path to a destination.

A BGP4 route consists of the following information:

- Network number (prefix) – A value comprised of the network mask bits and an IP address (*IP-address/ mask-bits*); for example, 192.168.129.0/18 indicates a network mask of 18 bits applied to the IP address 192.168.129.0. When a BGP4 Layer 3 switch advertises a route to one of its neighbors, the route is expressed in this format.
- AS-path – A list of the other autonomous systems through which a route passes. BGP4 routers can use the AS-path to detect and eliminate routing loops. For example, if a route received by a BGP4 router contains the AS that the router is in, the router does not add the route to its own BGP4 table. (The BGP4 RFCs refer to the AS-path as “AS_PATH”.)
- Additional path attributes – A list of additional parameters that describe the route. The route origin and next hop are examples of these additional path attributes.

NOTE
The Layer 3 switch re-advertises a learned best BGP4 route to the Layer 3 switch neighbors even when the software does not select that route for installation in the IP route table. The best BGP4 route is the route that the software selects based on comparison of the BGP4 route path attributes.

After a Brocade Layer 3 switch successfully negotiates a BGP4 session with a neighbor (a BGP4 peer), the Brocade Layer 3 switch exchanges complete BGP4 route tables with the neighbor. After this initial exchange, the Brocade Layer 3 switch and all other RFC 1771-compliant BGP4 routers send UPDATE messages to inform neighbors of new, changed, or no longer feasible routes. BGP4 routers do not send regular updates. However, if configured to do so, a BGP4 router does regularly send KEEPALIVE messages to its peers to maintain BGP4 sessions with them if the router does not have any route information to send in an UPDATE message.Refer to “BGP4 message types” on page 285 for information about BGP4 messages.

How BGP4 selects a path for a route

When multiple paths for the same route are known to a BGP4 router, the router uses the following algorithm to weigh the paths and determine the optimal path for the route. The optimal path depends on various parameters, which can be modified. (Refer to “Optional BGP4 configuration tasks” on page 304.)

1. Is the next hop accessible though an Interior Gateway Protocol (IGP) route? If not, ignore the route.

   NOTE
   The device does not use the default route to resolve BGP4 next hop. Also refer to “Enabling next-hop recursion” on page 310.

2. Use the path with the largest weight.

3. If the weights are the same, prefer the route with the largest local preference.
4. If the routes have the same local preference, prefer the route that was originated locally (by this BGP4 Layer 3 switch).
5. If the local preferences are the same, prefer the route with the shortest AS-path. An AS-SET counts as 1. A confederation path length, if present, is not counted as part of the path length.
6. If the AS-path lengths are the same, prefer the route with the lowest origin type. From low to high, route origin types are valued as follows:
   - IGP is lowest
   - EGP is higher than IGP but lower than INCOMPLETE
   - INCOMPLETE is highest
7. If the routes have the same origin type, prefer the route with the lowest MED. For a definition of MED, refer to "Configuring the Layer 3 switch to always compare Multi-Exit Discriminators" on page 316.

   BGP4 compares the MEDs of two otherwise equivalent paths if and only if the routes were learned from the same neighboring AS. This behavior is called deterministic MED. Deterministic MED is always enabled and cannot be disabled. In addition, you can enable the Layer 3 switch to always compare the MEDs, regardless of the AS information in the paths. To enable this comparison, enter the **always-compare-med** command at the BGP4 configuration level of the CLI. This option is disabled by default.

   NOTE
   By default, value 0 (most favorable) is used in MED comparison when the MED attribute is not present. The default MED comparison results in the Layer 3 switch favoring the route paths that are missing their MEDs. You can use the **med-missing-as-worst** command to make the Layer 3 switch regard a BGP route with a missing MED attribute as the least favorable route, when comparing the MEDs of the routes.

   NOTE
   MED comparison is not performed for internal routes originated within the local AS or confederation.

8. Prefer routes in the following order:
   - Routes received through EBGP from a BGP4 neighbor outside of the confederation
   - Routes received through EBGP from a BGP4 router within the confederation
   - Routes received through IBGP
9. If all the comparisons above are equal, prefer the route with the lowest IGP metric to the BGP4 next hop. This is the closest internal path inside the AS to reach the destination.
10. If the internal paths also are the same and BGP4 load sharing is enabled, load share among the paths. Otherwise, prefer the route that comes from the BGP4 router with the lowest router ID.

NOTE
Brocade Layer 3 switches support BGP4 load sharing among multiple equal-cost paths. BGP4 load sharing enables the Layer 3 switch to balance the traffic across the multiple paths instead of choosing just one path based on router ID. For EBGP routes, load sharing applies only when the paths are from neighbors within the same remote AS. EBGP paths from neighbors in different autonomous systems are not compared.

BGP4 message types

BGP4 routers communicate with their neighbors (other BGP4 routers) using the following types of messages:

- OPEN
- UPDATE
- KEEPALIVE
- NOTIFICATION

OPEN messages exchanged with BGP4 routers

After a BGP4 router establishes a TCP connection with a neighboring BGP4 router, the routers exchange OPEN messages. An OPEN message indicates the following:

- **BGP version** – Indicates the version of the protocol that is in use on the router. BGP version 4 supports Classless Interdomain Routing (CIDR) and is the version most widely used in the Internet. Version 4 also is the only version supported on Brocade Layer 3 switches.
- **AS number** – A two-byte number that identifies the AS to which the BGP4 router belongs.
- **Hold Time** – The number of seconds a BGP4 router will wait for an UPDATE or KEEPALIVE message (described below) from a BGP4 neighbor before assuming that the neighbor is dead. BGP4 routers exchange UPDATE and KEEPALIVE messages to update route information and maintain communication. If BGP4 neighbors are using different Hold Times, the lowest Hold Time is used by the neighbors. If the Hold Time expires, the BGP4 router closes its TCP connection to the neighbor and clears any information it has learned from the neighbor and cached.

  You can configure the Hold Time to be 0, in which case a BGP4 router will consider its neighbors to always be up. For directly-attached neighbors, you can configure the Brocade Layer 3 switch to immediately close the TCP connection to the neighbor and clear entries learned from an EBGP neighbor if the interface to that neighbor goes down. This capability is provided by the fast external fallover feature, which is disabled by default.
- **BGP Identifier** – The router ID. The BGP Identifier (router ID) identifies the BGP4 router to other BGP4 routers. Brocade Layer 3 switches use the same router ID for OSPF and BGP4. If you do not set a router ID, the software uses the IP address on the lowest numbered loopback interface configured on the router. If the Layer 3 switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, refer to “Changing the router ID” on page 31.
- **Parameter list** – An optional list of additional parameters used in peer negotiation with BGP4 neighbors.

UPDATE messages from BGP4 routers

After BGP4 neighbors establish a BGP4 connection over TCP and exchange their BGP4 routing tables, they do not send periodic routing updates. Instead, a BGP4 neighbor sends an update to its neighbor when it has a new route to advertise or routes have changed or become unfeasible. An UPDATE message can contain the following information:

- Network Layer Reachability Information (NLRI) – The mechanism by which BGP4 supports Classless Interdomain Routing (CIDR). An NLRI entry consists of an IP prefix that indicates a network being advertised by the UPDATE message. The prefix consists of an IP network number and the length of the network portion of the number. For example, an UPDATE message with the NLRI entry 192.168.129.0/18 indicates a route to IP network 192.168.129.0 with network mask 255.255.192.0. The binary equivalent of this mask is 18 consecutive one bits, thus “18” in the NLRI entry.
- Path attributes – Parameters that indicate route-specific information such as path information, route preference, next hop values, and aggregation information. BGP4 uses the path attributes to make filtering and routing decisions.
- Unreachable routes – A list of routes that have been in the sending router BGP4 table but are no longer feasible. The UPDATE message lists unreachable routes in the same format as new routes.

KEEPALIVE messages from BGP4 routers

BGP4 routers do not regularly exchange UPDATE messages to maintain the BGP4 sessions. For example, if a Layer 3 switch configured to perform BGP4 routing has already sent the latest route information to its peers in UPDATE messages, the router does not send more UPDATE messages. Instead, BGP4 routers send KEEPALIVE messages to maintain the BGP4 sessions. KEEPALIVE messages are 19 bytes long and consist only of a message header; they contain no routing data.

BGP4 routers send KEEPALIVE messages at a regular interval, the Keep Alive Time. The default Keep Alive Time on Brocade Layer 3 switches is 60 seconds.

A parameter related to the Keep Alive Time is the Hold Time. A BGP4 router Hold Time determines how many seconds the router will wait for a KEEPALIVE or UPDATE message from a BGP4 neighbor before deciding that the neighbor is dead. The Hold Time is negotiated when BGP4 routers exchange OPEN messages; the lower Hold Time is then used by both neighbors. For example, if BGP4 Router A sends a Hold Time of 5 seconds and BGP4 Router B sends a Hold Time of 4 seconds, both routers use 4 seconds as the Hold Time for their BGP4 session. The default Hold Time is 180 seconds. Generally, the Hold Time is configured to three times the value of the Keep Alive Time.

If the Hold Time is 0, a BGP4 router assumes that its neighbor is alive regardless of how many seconds pass between receipt of UPDATE or KEEPALIVE messages.

NOTIFICATION messages from BGP4 routers

When you close the router BGP4 session with a neighbor, or the router detects an error in a message received from the neighbor, or an error occurs on the router, the router sends a NOTIFICATION message to the neighbor. No further communication takes place between the BGP4 router that sent the NOTIFICATION and the neighbors that received the NOTIFICATION.

BGP4 graceful restart

BGP4 graceful restart is a high-availability routing feature that minimizes disruption in traffic forwarding, diminishes route flapping, and provides continuous service during a system restart. During such events, routes remain available between devices. BGP4 graceful restart operates between a device and its peers, and must be configured on each participating device.

Under normal operation, when a BGP4 device is restarted, the network is automatically reconfigured. Routes available through the restarting device are deleted when the device goes down, and are then rediscovered and added back to the routing tables when the device is back up and running. In a network with devices that are regularly restarted, performance can degrade significantly and the availability of network resources can be limited.

BGP4 graceful restart is enabled globally by default. A BGP4 graceful restart-enabled device advertises this capability to establish peering relationships with other devices. When a restart begins, neighbor devices mark all of the routes from the restarting device as stale, but continue to use the routes for the length of time specified by the restart timer. After the device is restarted, it begins to receive routing updates from the peers. When it receives the end-of-RIB marker that indicates it has received all of the BGP4 route updates, it recomputes the new routes and replaces the stale routes in the route map with the newly computed routes. If the device does not come back up within the time configured for the purge timer, the stale routes are removed.

This implementation of BGP4 graceful restart supports the Internet Draft-ietf-idr-restart-10.txt: restart mechanism for BGP4

For details concerning configuration of BGP4 graceful restart, refer to "Configuring BGP4 graceful restart" on page 324.

Basic configuration and activation for BGP4

BGP4 is disabled by default. Follow the steps below to enable BGP4 and place your Brocade Layer 3 switch into service as a BGP4 router.

1. Enable the BGP4 protocol.
2. Set the local AS number.

   NOTE
   You must specify the local AS number for BGP4 to become functional.

3. Add each BGP4 neighbor (peer BGP4 router) and identify the AS the neighbor is in.
4. Save the BGP4 configuration information to the system configuration file.

NOTE
By default, the Brocade router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 switch does not have a loopback interface, the default router ID is the lowest numbered IP interface address configured on the device. For more information or to change the router ID, refer to "Changing the router ID" on page 31. If you change the router ID, all current BGP4 sessions are cleared.

```
Brocade> enable
Brocade#configure terminal
Brocade(config)#router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
Brocade(config-bgp-router)#local-as 10
Brocade(config-bgp-router)#neighbor 192.168.23.99 remote-as 100
Brocade(config-bgp-router)#write memory
```

NOTE

When BGP4 is enabled on a Brocade Layer 3 switch, you do not need to reset the system. The protocol is activated as soon as you enable it. Moreover, the router begins a BGP4 session with a BGP4 neighbor as soon as you add the neighbor.

Note regarding disabling BGP4

If you disable BGP4, the Layer 3 switch removes all the running configuration information for the disabled protocol from the running-config. To restore the BGP4 configuration, you must reload the software to load the configuration from the startup-config. Moreover, when you save the configuration to the startup-config file after disabling the protocol, all the configuration information for the disabled protocol is removed from the startup-config file.

The CLI displays a warning message such as the following.

```
Brocade(config-bgp-router)#no router bgp
router bgp mode now disabled. All bgp config data will be lost when writing to
flash!
```

If you are testing a BGP4 configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

NOTE

To disable BGP4 without losing the BGP4 configuration information, remove the local AS (for example, by entering the **no local-as** *num* command). In this case, BGP4 retains the other configuration information but is not operational until you set the local AS again.

BGP4 parameters

You can modify or set the following BGP4 parameters:

- Optional – Define the router ID. (The same router ID also is used by OSPF.)
- Required – Specify the local AS number.
- Optional – Add a loopback interface for use with neighbors.
- Required – Identify BGP4 neighbors.
- Optional – Change the Keep Alive Time and Hold Time.
- Optional – Change the update timer for route changes.
- Optional – Enable fast external fallover.

- Optional – Specify a list of individual networks in the local AS to be advertised to remote autonomous systems using BGP4.
- Optional – Change the default local preference for routes.
- Optional – Enable the default route (default-information-originate).
- Optional – Enable use of a default route to resolve a BGP4 next-hop route.
- Optional – Change the default MED (metric).
- Optional – Enable next-hop recursion.
- Optional – Change the default administrative distances for EBGP, IBGP, and locally originated routes.
- Optional – Require the first AS in an Update from an EBGP neighbor to be the neighbor AS.
- Optional – Change MED comparison parameters.
- Optional – Disable comparison of the AS-Path length.
- Optional – Enable comparison of the router ID.
- Optional – Enable auto summary to summarize routes at an IP class boundary (A, B, or C).
- Optional – Aggregate routes in the BGP4 route table into CIDR blocks.
- Optional – Configure the router as a BGP4 router reflector.
- Optional – Configure the Layer 3 switch as a member of a BGP4 confederation.
- Optional – Change the default metric for routes that BGP4 redistributes into RIP or OSPF.
- Optional – Change the parameters for RIP, OSPF, or static routes redistributed into BGP4.
- Optional – Change the number of paths for BGP4 load sharing.
- Optional – Change other load-sharing parameters
- Optional – Define BGP4 address filters.
- Optional – Define BGP4 AS-path filters.
- Optional – Define BGP4 community filters.
- Optional – Define IP prefix lists.
- Optional – Define neighbor distribute lists.
- Optional – Define BGP4 route maps for filtering routes redistributed into RIP and OSPF.
- Optional – Define route flap dampening parameters.

NOTE
When using the CLI, you set global level parameters at the BGP CONFIG level of the CLI. You can reach the BGP CONFIG level by entering **router bgp…** at the global CONFIG level.

BGP4 parameter changes

Some parameter changes take effect immediately while others do not take full effect until the router sessions with its neighbors are reset. Some parameters do not take effect until the router is rebooted.

Parameter changes that take effect immediately

- Enable or disable BGP.
- Set or change the local AS.

- Add neighbors.
- Change the update timer for route changes.
- Disable or enable fast external fallover.
- Specify individual networks that can be advertised.
- Change the default local preference, default information originate setting, or administrative distance.
- Enable or disable use of a default route to resolve a BGP4 next-hop route.
- Enable or disable MED (metric) comparison.
- Require the first AS in an Update from an EBGP neighbor to be the neighbor AS.
- Change MED comparison parameters.
- Disable comparison of the AS-Path length.
- Enable comparison of the router ID.
- Enable next-hop recursion.
- Enable or disable auto summary.
- Change the default metric.
- Disable or re-enable route reflection.
- Configure confederation parameters.
- Disable or re-enable load sharing.
- Change the maximum number of load-sharing paths.
- Change other load-sharing parameters.
- Define route flap dampening parameters.
- Add, change, or negate redistribution parameters (except changing the default MED; see below).
- Add, change, or negate route maps (when used by the **network** command or a redistribution command).

BGP4 parameter changes after resetting neighbor sessions

The following parameter changes take effect only after the router BGP4 sessions are cleared, or reset using the “soft” clear option. (Refer to “Closing or resetting a neighbor session” on page 396.)

The parameter are as follows:

- Change the Hold Time or Keep Alive Time.
- Aggregate routes.
- Add, change, or negate filter tables.

BGP4 parameter changes after disabling and re-enabling redistribution

The following parameter change takes effect only after you disable and then re-enable redistribution:

- Change the default MED (metric).

Basic configuration tasks required for BGP4

The following sections describe how to perform the configuration tasks that are required to use BGP4 on the Brocade Layer 3 switch. You can modify many parameters in addition to the ones described in this section. Refer to "Optional BGP4 configuration tasks" on page 304.

Enabling BGP4 on the router

When you enable BGP4 on the router, BGP4 is automatically activated. To enable BGP4 on the router, enter the following commands.

```
Brocade> enable
Brocade#configure terminal
Brocade(config)#router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
Brocade(config-bgp-router)#local-as 10
Brocade(config-bgp-router)#neighbor 192.168.23.99 remote-as 100
Brocade(config-bgp-router)#write memory
```

Changing the router ID

The OSPF and BGP4 protocols use router IDs to identify the routers that are running the protocols. A router ID is a valid, unique IP address and sometimes is an IP address configured on the router. The router ID cannot be an IP address in use by another device.

By default, the router ID on a Brocade Layer 3 switch is one of the following:

- If the router has loopback interfaces, the default router ID is the IP address configured on the lowest numbered loopback interface configured on the Layer 3 switch. For example, if you configure loopback interfaces 1, 2, and 3 as follows, the default router ID is 10.9.9.9/24:
  - Loopback interface 1, 10.9.9.9/24
  - Loopback interface 2, 10.4.4.4/24
  - Loopback interface 3, 10.1.1.1/24
- If the device does not have any loopback interfaces, the default router ID is the lowest numbered IP interface address configured on the device.

NOTE
Brocade Layer 3 switches use the same router ID for both OSPF and BGP4. If the router is already configured for OSPF, you may want to use the router ID that is already in use on the router rather than set a new one. To display the router ID, enter the **show ip** CLI command at any CLI level.

To change the router ID, enter a command such as the following.

```
Brocade(config)#ip router-id 192.168.22.26
```

Syntax: **ip router-id** *ip-addr*

The *ip-addr* can be any valid, unique IP address.

NOTE
You can specify an IP address used for an interface on the Brocade Layer 3 switch, but do not specify an IP address in use by another device.

Setting the local AS number

The local AS number identifies the AS the Brocade BGP4 router is in. The AS number can be from 1 through 65535. There is no default. AS numbers 64512 through 65535 are the well-known private BGP4 AS numbers and are not advertised to the Internet community.

To set the local AS number, enter commands such as the following.

```
Brocade(config)#router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
Brocade(config-bgp-router)#local-as 10
Brocade(config-bgp-router)#write memory
```

Syntax: [no] local-as *num*

The *num* parameter specifies the local AS number.

Adding a loopback interface

You can configure the router to use a loopback interface instead of a specific port or virtual routing interface to communicate with a BGP4 neighbor. A loopback interface adds stability to the network by working around route flap problems that can occur due to unstable links between the router and its neighbors.

Loopback interfaces are always up, regardless of the states of physical interfaces. Loopback interfaces are especially useful for IBGP neighbors (neighbors in the same AS) that are multiple hops away from the router. When you configure a BGP4 neighbor on the router, you can specify whether the router uses the loopback interface to communicate with the neighbor. As long as a path exists between the router and its neighbor, BGP4 information can be exchanged. The BGP4 session is not associated with a specific link but instead is associated with the virtual interfaces.

You can add up to 24 IP addresses to each loopback interface.

NOTE
If you configure the Brocade Layer 3 switch to use a loopback interface to communicate with a BGP4 neighbor, the peer IP address on the remote router pointing to your loopback address must be configured.

To add a loopback interface, enter commands such as those shown in the following example.

```
Brocade(config-bgp-router)#exit
Brocade(config)#int loopback 1
Brocade(config-lbif-1)#ip address 10.0.0.1/24
```

Syntax: interface loopback *num*

The *num* value can be from 1 through 8 on Chassis Layer 3 switches. The value can be from 1 through 4 on the Compact Layer 3 switch.

Adding BGP4 neighbors

The BGP4 protocol does not contain a peer discovery process. Therefore, for each of the router BGP4 neighbors (peers), you must indicate the neighbor IP address and the AS each neighbor is in. Neighbors that are in different autonomous systems communicate using EBGP. Neighbors within the same AS communicate using IBGP.

NOTE
If the Layer 3 switch has multiple neighbors with similar attributes, you can simplify configuration by configuring a peer group, then adding individual neighbors to it. The configuration steps are similar, except you specify a peer group name instead of a neighbor IP address when configuring the neighbor parameters, then add individual neighbors to the peer group. Refer to "Adding a BGP4 peer group" on page 299.

NOTE
The Layer 3 switch attempts to establish a BGP4 session with a neighbor as soon as you enter a command specifying the neighbor IP address. If you want to completely configure the neighbor parameters before the Layer 3 switch establishes a session with the neighbor, you can administratively shut down the neighbor. Refer to "Administratively shutting down a session with a BGP4 neighbor" on page 302.

To add a BGP4 neighbor with IP address 192.168.22.26, enter the following command.

```
Brocade(config-bgp-router)#neighbor 192.168.22.26
```

The neighbor *ip-addr* must be a valid IP address.

The **neighbor** command has some additional parameters, as shown in the following syntax:

Syntax: [**no**] **neighbor** *ip-addr* | *peer-group-name*
[**advertisement-interval** *num*]
[**capability orf prefixlist** [**send** | **receive**]]
[**default-originate** [**route-map** *map-name*]]
[**description** *string*]
[**distribute-list in** | **out** *num,num,...* | *ACL-num* **in** | **out**]
[**ebgp-multihop** [*num*]]
[**filter-list in** | **out** *num,num,...* | *ACL-num* **in** | **out** | **weight**]
[**maximum-prefix** *num* [*threshold*] [**teardown**]]
[**next-hop-self**]
[**nlri multicast** | **unicast** | **multicast unicast**]
[**password** [**0** | **1**] *string*]
[**prefix-list** *string* **in** | **out**]
[**remote-as** *as-number*]
[**remove-private-as**]
[**route-map in** | **out** *map-name*]
[**route-reflector-client**]
[**send-community**]
[**soft-reconfiguration inbound**]
[**shutdown**]
[**timers keep-alive** *num* **hold-time** *num*]
[**unsuppress-map** *map-name*]
[**update-source** *ip-addr* | **ethernet** *port* | **loopback** *num* | **ve** *num*]
[**weight** *num*]

The *ip-addr* | *peer-group-name* parameter indicates whether you are configuring an individual neighbor or a peer group. If you specify a neighbor IP address, you are configuring that individual neighbor. If you specify a peer group name, you are configuring a peer group. Refer to "Adding a BGP4 peer group" on page 299.

advertisement-interval *num* specifies the minimum delay (in seconds) between messages to the specified neighbor. The default is 30 for EBGP neighbors (neighbors in other autonomous systems). The default is 5 for IBGP neighbors (neighbors in the same AS). The range is 0 through 600.

NOTE
The Layer 3 switch applies the advertisement interval only under certain conditions. The Layer 3 switch does not apply the advertisement interval when sending initial updates to a BGP4 neighbor. As a result, the Layer 3 switch sends the updates one immediately after another, without waiting for the advertisement interval.

capability orf prefixlist [**send** | **receive**] configures cooperative router filtering. The **send** | **receive** parameter specifies the support you are enabling:

- **send** – The Layer 3 switch sends the IP prefix lists as Outbound Route Filters (ORFs) to the neighbor.
- **receive** – The Layer 3 switch accepts filters as Outbound Route Filters (ORFs) from the neighbor.

If you do not specify the capability, both capabilities are enabled. The **prefixlist** parameter specifies the type of filter you want to send to the neighbor.

For more information, refer to “Configuring cooperative BGP4 route filtering” on page 351.

NOTE
The current release supports cooperative filtering only for filters configured using IP prefix lists.

default-originate [**route-map** *map-name*] configures the Layer 3 switch to send the default route 0.0.0.0 to the neighbor. If you use the route-map *map-name* parameter, the route map injects the default route conditionally, based on the match conditions in the route map.

description *string* specifies a name for the neighbor. You can enter an alphanumeric text string up to 80 characters long.

distribute-list in | **out** *num,num,...* specifies a distribute list to be applied to updates to or from the specified neighbor. The **in** | **out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. The *num,num,...* parameter specifies the list of address-list filters. The router applies the filters in the order in which you list them and stops applying the filters in the distribute list when a match is found.

Alternatively, you can specify **distribute-list** *ACL-num* **in** | **out** to use an IP ACL instead of a distribute list. In this case, *ACL-num* is an IP ACL.

NOTE
By default, if a route does not match any of the filters, the Layer 3 switch denies the route. To change the default behavior, configure the last filter as “permit any any”.

NOTE
The address filter must already be configured. Refer to “Specific IP address filtering” on page 333.

ebgp-multihop [*num*] specifies that the neighbor is more than one hop away and that the session type with the neighbor is thus EBGP-multihop. This option is disabled by default. The *num* parameter specifies the TTL you are adding for the neighbor. You can specify a number from 0 through 255. The default is 0. If you leave the EBGP TTL value set to 0, the software uses the IP TTL value.

filter-list in | out *num,num,..* specifies an AS-path filter list or a list of AS-path ACLs. The **in | out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. If you specify in or out, The *num,num,...* parameter specifies the list of AS-path filters. The router applies the filters in the order in which you list them and stops applying the filters in the AS-path filter list when a match is found. The **weight** *num* parameter specifies a weight that the Layer 3 switch applies to routes received from the neighbor that match the AS-path filter or ACL. You can specify a number from 0 through 65535.

Alternatively, you can specify filter-list *ACL-num* **in | out | weight** to use an AS-path ACL instead of an AS-path filter list. In this case, *ACL-num* is an AS-path ACL.

NOTE
By default, if an AS-path does not match any of the filters or ACLs, the Layer 3 switch denies the route. To change the default behavior, configure the last filter or ACL as "permit any any".

NOTE
The AS-path filter or ACL must already be configured. Refer to "AS-path filtering" on page 334.

maximum-prefix *num* specifies the maximum number of IP network prefixes (routes) that can be learned from the specified neighbor or peer group. You can specify a value from 0 through 4294967295. The default is 0 (unlimited):

- The *num* parameter specifies the maximum number. You can specify a value from 0 through 4294967295. The default is 0 (unlimited).
- The *threshold* parameter specifies the percentage of the value you specified for the **maximum-prefix** *num*, at which you want the software to generate a Syslog message. You can specify a value from 1 (one percent) to 100 (100 percent). The default is 100.
- The **teardown** parameter tears down the neighbor session if the maximum-prefix limit is exceeded. The session remains shutdown until you clear the prefixes using the **clear ip bgp neighbor all** or **clear ip bgp neighbor** *ip-addr* command, or change the neighbor maximum-prefix configuration. The software also generates a Syslog message.

next-hop-self specifies that the router should list itself as the next hop in updates sent to the specified neighbor. This option is disabled by default.

The **nlri multicast | unicast | multicast** unicast parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. Optionally, you also can specify unicast if you want the Layer 3 switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is unicast only.

password [0 | 1] *string* specifies an MD5 password for securing sessions between the Layer 3 switch and the neighbor. You can enter a string up to 80 characters long. The string can contain any alphanumeric characters, but the first character cannot be a number. If the password contains a number, do not enter a space following the number.

The **0 | 1** parameter is the encryption option, which you can omit (the default) or which can be one of the following:

- **0** – Disables encryption for the authentication string you specify with the command. The password or string is shown as clear text in the output of commands that display neighbor or peer group configuration information.
- **1** – Assumes that the authentication string you enter is the encrypted form, and decrypts the value before using it.

For more information, refer to "Encryption of BGP4 MD5 authentication keys" on page 297.

NOTE
If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter 0 or 1. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option 1, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option 1 followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

prefix-list *string* **in** | **out** specifies an IP prefix list. You can use IP prefix lists to control routes to and from the neighbor. IP prefix lists are an alternative method to AS-path filters. The **in** | **out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. You can configure up to 1000 prefix list filters. The filters can use the same prefix list or different prefix lists. To configure an IP prefix list, refer to "Defining IP prefix lists" on page 340.

remote-as *as-number* specifies the AS the remote neighbor is in. The *as-number* can be a number from 1 through 65535. There is no default.

remove-private-as configures the router to remove private AS numbers from UPDATE messages the router sends to this neighbor. The router will remove AS numbers 64512 through 65535 (the well-known BGP4 private AS numbers) from the AS-path attribute in UPDATE messages the Layer 3 switch sends to the neighbor. This option is disabled by default.

route-map in | **out** *map-name* specifies a route map the Layer 3 switch will apply to updates sent to or received from the specified neighbor. The **in** | **out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor.

NOTE
The route map must already be configured. Refer to "Defining route maps" on page 342.

route-reflector-client specifies that this neighbor is a route-reflector client of the router. Use the parameter only if this router is going to be a route reflector. For information, refer to "Route reflection parameter configuration" on page 317. This option is disabled by default.

send-community enables sending the community attribute in updates to the specified neighbor. By default, the router does not send the community attribute.

shutdown administratively shuts down the session with this neighbor. Shutting down the session allows you to completely configure the neighbor and save the configuration without actually establishing a session with the neighbor. This option is disabled by default.

soft-reconfiguration inbound enables the soft reconfiguration feature, which stores all the route updates received from the neighbor. If you request a soft reset of inbound routes, the software performs the reset by comparing the policies against the stored route updates, instead of requesting the neighbor BGP4 route table or resetting the session with the neighbor. Refer to "Using soft reconfiguration" on page 391.

timers keep-alive *num* **hold-time** *num* overrides the global settings for the Keep Alive Time and Hold Time. For the Keep Alive Time, you can specify from 0 through 65535 seconds. For the Hold Time, you can specify 0 or 3 through 65535 (1 and 2 are not allowed). If you set the Hold Time to 0, the router waits indefinitely for messages from a neighbor without concluding that the neighbor is dead. The defaults for these parameters are the currently configured global Keep Alive Time and Hold Time. For more information about these parameters, refer to "Changing the Keep Alive Time and Hold Time" on page 304.

unsuppress-map *map-name* removes route dampening from a neighbor routes when those routes have been dampened due to aggregation. Refer to “Removing route dampening from neighbor routes suppressed due to aggregation” on page 357.

update-source *ip-addr* | **ethernet** *port* | **loopback** *num* | **ve** *num* configures the router to communicate with the neighbor through the specified interface. There is no default.

weight *num* specifies a weight the Layer 3 switch will add to routes received from the specified neighbor. BGP4 prefers larger weights over smaller weights. The default weight is 0.

Encryption of BGP4 MD5 authentication keys

When you configure a BGP4 neighbor or neighbor peer group, you can specify an MD5 authentication string for authenticating packets exchanged with the neighbor or peer group of neighbors.

For added security, the software encrypts display of the authentication string by default. The software also provides an optional parameter to disable encryption of the authentication string, on an individual neighbor or peer group basis. By default, the MD5 authentication strings are displayed in encrypted format in the output of the following commands:

- **show running-config** (or **write terminal**)
- **show configuration**
- **show ip bgp config**

When encryption of the authentication string is enabled, the string is encrypted in the CLI regardless of the access level you are using.

If you display the running-config after reloading, the BGP4 commands that specify an authentication string show the string in encrypted form.

In addition, when you save the configuration to the startup-config file, the file contains the new BGP4 command syntax and encrypted passwords or strings.

NOTE
Brocade recommends that you save a copy of the startup-config file for each switch you plan to upgrade.

Encryption example

The following commands configure a BGP4 neighbor and a peer group, and specify MD5 authentication strings (passwords) for authenticating packets exchanged with the neighbor or peer group.

```
Brocade(config-bgp-router)#local-as 2
Brocade(config-bgp-router)#neighbor xyz peer-group
Brocade(config-bgp-router)#neighbor xyz password abc
Brocade(config-bgp-router)#neighbor 10.10.200.102 peer-group xyz
Brocade(config-bgp-router)#neighbor 10.10.200.102 password test
```

Here is how the commands appear when you display the BGP4 configuration commands.

```
Brocade#show ip bgp config
Current BGP configuration:
router bgp
 local-as 2
 neighbor xyz peer-group
 neighbor xyz password 1 $!2d
 neighbor 10.10.200.102 peer-group xyz
 neighbor 10.10.200.102 remote-as 1
 neighbor 10.10.200.102 password 1 $on-o
```

Notice that the software has converted the commands that specify an authentication string into the new syntax (described below), and has encrypted display of the authentication strings.

Command syntax

Since the default behavior does not affect the BGP4 configuration itself but does encrypt display of the authentication string, the CLI does not list the encryption options.

Syntax: [no] neighbor *ip-addr* | *peer-group-name* password [0 | 1] *string*

The *ip-addr* | *peer-group-name* parameter indicates whether you are configuring an individual neighbor or a peer group. If you specify a neighbor IP address, you are configuring that individual neighbor. If you specify a peer group name, you are configuring a peer group.

The password *string* parameter specifies an MD5 authentication string for securing sessions between the Layer 3 switch and the neighbor. You can enter a string up to 80 characters long. The string can contain any alphanumeric characters, but the first character cannot be a number. If the password contains a number, do not enter a space following the number.

The 0 | 1 parameter is the encryption option, which you can omit (the default) or which can be one of the following:

- 0 – Disables encryption for the authentication string you specify with the command. The password or string is shown as clear text in the output of commands that display neighbor or peer group configuration information.
- 1 – Assumes that the authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE
If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter 0 or 1. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option 1, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option 1 followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

Displaying the Authentication String

If you want to display the authentication string, enter the following commands.

```
Brocade(config)#enable password-display
Brocade#show ip bgp neighbors
```

The **enable password-display** command enables display of the authentication string, but only in the output of the **show ip bgp neighbors** command. Display of the string is still encrypted in the startup-config file and running-config. Enter the command at the global CONFIG level of the CLI.

NOTE
The command also displays SNMP community strings in clear text, in the output of the **show snmp server** command.

Adding a BGP4 peer group

A *peer group* is a set of BGP4 neighbors that share common parameters. Peer groups provide the following benefits:

- **Simplified neighbor configuration** – You can configure a set of neighbor parameters and then apply them to multiple neighbors. You do not need to individually configure the common parameters individually on each neighbor.
- **Flash memory conservation** – Using peer groups instead of individually configuring all the parameters for each neighbor requires fewer configuration commands in the startup-config file.

You can perform the following tasks on a peer-group basis:

- Reset neighbor sessions
- Perform soft-outbound resets (the Layer 3 switch updates outgoing route information to neighbors but does not entirely reset the sessions with those neighbors)
- Clear BGP message statistics
- Clear error buffers

Peer group parameters

You can set all neighbor parameters in a peer group. When you add a neighbor to the peer group, the neighbor receives all the parameter settings you set in the group, except parameter values you have explicitly configured for the neighbor. If you do not set a neighbor parameter in the peer group and the parameter also is not set for the individual neighbor, the neighbor uses the default value.

Peer group configuration rules

The following rules apply to peer group configuration:

- You must configure a peer group before you can add neighbors to the peer group.
- If you remove a parameter from a peer group, the value for that parameter is reset to the default for all the neighbors within the peer group, unless you have explicitly set that parameter on individual neighbors. In this case, the value you set on the individual neighbors applies to those neighbors, while the default value applies to neighbors for which you have not explicitly set the value.

NOTE
If you enter a command to remove the remote AS parameter from a peer group, the software checks to ensure that the peer group does not contain any neighbors. If the peer group does contain neighbors, the software does not allow you to remove the remote AS. The software prevents removing the remote AS in this case so that the neighbors in the peer group that are using the remote AS do not lose connectivity to the Layer 3 switch.

- Once you add a neighbor to a peer group, you cannot configure the following outbound parameters (the parameters governing outbound traffic) for the neighbor:
    - Default-information-originate
    - Next-hop-self
    - Outbound route map
    - Outbound filter list
    - Outbound distribute list
    - Outbound prefix list
    - Remote AS, if configured for the peer group
    - Remove private AS
    - Route reflector client
    - Send community
    - Timers
    - Update source

    If you want to change an outbound parameter for an individual neighbor, you must first remove the neighbor from the peer group. In this case, you cannot re-add the neighbor to the same peer group, but you can add the neighbor to a different peer group. All the neighbors within a peer group must have the same values for the outbound parameters. To change an outbound parameter to the same value for all neighbors within a peer group, you can change the parameter on a peer-group basis. In this case, you do not need to remove the neighbors and change the parameter individually for each neighbor.
- If you add an outbound parameter to a peer group, that parameter is automatically applied to all neighbors within the peer group.
- When you add a neighbor to a peer group, the software removes any outbound parameters for that neighbor from the running configuration (running-config). As a result, when you save the configuration to the startup-config file, the file does not contain any outbound parameters for the individual neighbors you have placed in a peer group. The only outbound parameters the startup-config file contains for neighbors within a peer group are the parameters associated with the peer group itself. However, the running-config and the startup-config file can contain individual parameters listed in the previous section as well as the settings for those parameters within a peer group.

You can override neighbor parameters that do not affect outbound policy on an individual neighbor basis.

- If you do not specify a parameter for an individual neighbor, the neighbor uses the value in the peer group.
- If you set the parameter for the individual neighbor, that value overrides the value you set in the peer group.

- If you add a parameter to a peer group that already contains neighbors, the parameter value is applied to neighbors that do not already have the parameter explicitly set. If a neighbor has the parameter explicitly set, the explicitly set value overrides the value you set for the peer group.
- If you remove the setting for a parameter from a peer group, the value for that parameter changes to the default value for all the neighbors in the peer group that do not have that parameter individually set.

Configuring a peer group

To configure a BGP4 peer group, enter commands such as the following at the BGP configuration level.

```
Brocade(config-bgp-router)#neighbor PeerGroup1 peer-group
Brocade(config-bgp-router)#neighbor PeerGroup1 description “EastCoast Neighbors”
Brocade(config-bgp-router)#neighbor PeerGroup1 remote-as 100
Brocade(config-bgp-router)#neighbor PeerGroup1 distribute-list out 1
```

The commands in this example configure a peer group called “PeerGroup1” and set the following parameters for the peer group:

- A description, “EastCoast Neighbors”
- A remote AS number, 100
- A distribute list for outbound traffic

The software applies these parameters to each neighbor you add to the peer group. You can override the description parameter for individual neighbors. If you set the description parameter for an individual neighbor, the description overrides the description configured for the peer group. However, you cannot override the remote AS and distribute list parameters for individual neighbors. Since these parameters control outbound traffic, the parameters must have the same values for all neighbors within the peer group.

Syntax: **neighbor** *peer-group-name* **peer-group**

The *peer-group-name* parameter specifies the name of the group and can be up to 80 characters long. The name can contain special characters and internal blanks. If you use internal blanks, you must use quotation marks around the name. For example, the command **neighbor “My Three Peers” peer-group** is valid, but the command **neighbor My Three Peers peer-group** is not valid.

Syntax: [**no**] **neighbor** *ip-addr* | *peer-group-name*
[**advertisement-interval** *num*]
[**default-originate** [**route-map** *map-name*]]
[**description** *string*]
[**distribute-list in** | **out** *num,num,...* | *ACL-num* **in** | **out**]
[**ebgp-multihop** [*num*]]
[**filter-list in** | **out** *num,num,...* | *ACL-num* **in** | **out** | **weight**]
[**maximum-prefix** *num* [*threshold*] [**teardown**]]
[**next-hop-self**]
[**password** [**0** | **1**] *string*]
[**prefix-list** *string* **in** | **out**]
[**remote-as** *as-number*]
[**remove-private-as**]
[**route-map in** | **out** *map-name*]
[**route-reflector-client**]
[**send-community**]

[**soft-reconfiguration inbound**]
[**shutdown**]
[**timers keep-alive** *num* **hold-time** *num*]
[**update-source loopback** *num*]
[**weight** *num*]

The *ip-addr* | *peer-group-name* parameter indicates whether you are configuring a peer group or an individual neighbor. You can specify a peer group name or IP address with the **neighbor** command. If you specify a peer group name, you are configuring a peer group. If you specify a neighbor IP address, you are configuring that individual neighbor. Use the *ip-addr* parameter if you are configuring an individual neighbor instead of a peer group. Refer to "Adding BGP4 neighbors" on page 292.

The remaining parameters are the same ones supported for individual neighbors. Refer to "Adding BGP4 neighbors" on page 292.

Applying a peer group to a neighbor

After you configure a peer group, you can add neighbors to the group. When you add a neighbor to a peer group, you are applying all the neighbor attributes specified in the peer group to the neighbor.

To add neighbors to a peer group, enter commands such as the following.

```
Brocade(config-bgp-router)#neighbor 192.168.1.12 peer-group PeerGroup1
Brocade(config-bgp-router)#neighbor 192.168.2.45 peer-group PeerGroup1
Brocade(config-bgp-router)#neighbor 192.168.3.69 peer-group PeerGroup1
```

The commands in this example add three neighbors to the peer group "PeerGroup1". As members of the peer group, the neighbors automatically receive the neighbor parameter values configured for the peer group. You also can override the parameters (except parameters that govern outbound traffic) on an individual neighbor basis. For neighbor parameters not specified for the peer group, the neighbors use the default values.

Syntax: **neighbor** *ip-addr* **peer-group** *peer-group-name*

The *ip-addr* parameter specifies the IP address of the neighbor.

The *peer-group-name* parameter specifies the peer group name.

NOTE
You must add the peer group before you can add neighbors to it.

Administratively shutting down a session with a BGP4 neighbor

You can prevent the Layer 3 switch from starting a BGP4 session with a neighbor by administratively shutting down the neighbor. This option is very useful for situations in which you want to configure parameters for a neighbor but are not ready to use the neighbor. You can shut the neighbor down as soon as you have added it the Layer 3 switch, configure the neighbor parameters, then allow the Layer 3 switch to re-establish a session with the neighbor by removing the shutdown option from the neighbor.

When you apply the new option to shut down a neighbor, the option takes place immediately and remains in effect until you remove the option. If you save the configuration to the startup-config file, the shutdown option remains in effect even after a software reload.

NOTE
The software also contains an option to end the session with a BGP4 neighbor and thus clear the routes learned from the neighbor. Unlike this clear option, the option for shutting down the neighbor can be saved in the startup-config file and thus can prevent the Layer 3 switch from establishing a BGP4 session with the neighbor even after reloading the software.

NOTE
If you notice that a particular BGP4 neighbor never establishes a session with the Brocade Layer 3 switch, check the Layer 3 switch running-config and startup-config files to see whether the configuration contains a command that is shutting down the neighbor. The neighbor may have been shut down previously by an administrator.

To shut down a BGP4 neighbor, enter commands such as the following.

```
Brocade(config)#router bgp
Brocade(config-bgp-router)#neighbor 192.168.22.26 shutdown
Brocade(config-bgp-router)#write memory
```

Syntax: [**no**] **neighbor** *ip-addr* **shutdown**

The *ip-addr* parameter specifies the IP address of the neighbor.

Optional BGP4 configuration tasks

The following sections describe how to perform optional BGP4 configuration tasks.

Changing the Keep Alive Time and Hold Time

The Keep Alive Time specifies how frequently the router will send KEEPALIVE messages to its BGP4 neighbors. The Hold Time specifies how long the router will wait for a KEEPALIVE or UPDATE message from a neighbor before concluding that the neighbor is dead. When the router concludes that a BGP4 neighbor is dead, the router ends the BGP4 session and closes the TCP connection to the neighbor.

The default Keep Alive time is 60 seconds. The default Hold Time is 180 seconds. To change the timers, use either of the following methods.

NOTE
Generally, you should set the Hold Time to three times the value of the Keep Alive Time.

NOTE
You can override the global Keep Alive Time and Hold Time on individual neighbors. Refer to "Adding BGP4 neighbors" on page 292.

To change the Keep Alive Time to 30 and Hold Time to 90, enter the following command.

```
Brocade(config-bgp-router)#timers keep-alive 30 hold-time 90
```

Syntax: **timers keep-alive** *num* **hold-time** *num*

For each keyword, *num* indicates the number of seconds. The Keep Alive Time can be 0 through 65535. The Hold Time can be 0 or 3 through 65535 (1 and 2 are not allowed). If you set the Hold Time to 0, the router waits indefinitely for messages from a neighbor without concluding that the neighbor is dead.

Changing the BGP4 next-hop update timer

By default, the Layer 3 switch updates its BGP4 next-hop tables and affected BGP4 routes five seconds after IGP route changes. You can change the update timer to a value from 1 through 30 seconds.

To change the BGP4 update timer value, enter the **update-time** command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#update-time 15
```

This command changes the update timer to 15 seconds.

Syntax: [**no**] **update-time** *secs*

The *secs* parameter specifies the number of seconds and can be from 1 through 30. The default is 5.

Enabling fast external fallover

BGP4 routers rely on KEEPALIVE and UPDATE messages from neighbors to signify that the neighbors are alive. For BGP4 neighbors that are two or more hops away, such messages are the only indication that the BGP4 protocol has concerning the alive state of the neighbors. As a result, if a neighbor dies, the router will wait until the Hold Time expires before concluding that the neighbor is dead and closing its BGP4 session and TCP connection with the neighbor.

The router waits for the Hold Time to expire before ending the connection to a directly-attached BGP4 neighbor that dies.

For directly attached neighbors, the router immediately senses loss of a connection to the neighbor from a change of state of the port or interface that connects the router to its neighbor. For directly attached EBGP neighbors, the router can use this information to immediately close the BGP4 session and TCP connection to locally attached neighbors that die.

NOTE
The fast external fallover feature applies only to directly attached EBGP neighbors. The feature does not apply to IBGP neighbors.

If you want to enable the router to immediately close the BGP4 session and TCP connection to locally attached neighbors that die, use either of the following methods.

To enable fast external fallover, enter the following command.

```
Brocade(config-bgp-router)#fast-external-fallover
```

To disable fast external fallover again, enter the following command.

```
Brocade(config-bgp-router)#no fast-external-fallover
```

Syntax: [**no**] **fast-external-fallover**

Changing the maximum number of paths for BGP4 load sharing

Load sharing enables the Layer 3 switch to balance traffic to a route across multiple equal-cost paths of the same type (EBGP or IBGP) for the route.

To configure the Layer 3 switch to perform BGP4 load sharing:

- Enable IP load sharing if it is disabled.
- Set the maximum number of paths. The default maximum number of BGP4 load sharing paths is 1, which means no BGP4 load sharing takes place by default.

  NOTE
  The maximum number of BGP4 load sharing paths cannot be greater than the maximum number of IP load sharing paths.

How load sharing affects route selection

During evaluation of multiple paths to select the best path to a given destination for installment in the IP route table, the last comparison the Layer 3 switch performs is a comparison of the internal paths:

- When IP load sharing is disabled, the Layer 3 switch prefers the path to the router with the lower router ID.
- When IP load sharing and BGP4 load sharing are enabled, the Layer 3 switch balances the traffic across the multiple paths instead of choosing just one path based on router ID.

Refer to "How BGP4 selects a path for a route" on page 283 for a description of the BGP4 algorithm.

When you enable IP load sharing, the Layer 3 switch can load balance BGP4 or OSPF routes across up to four equal paths by default. You can change the number of IP load sharing paths to a value from 2 through 6.

How load sharing works

Load sharing is performed in round-robin fashion and is based on the destination IP address only. The first time the router receives a packet destined for a specific IP address, the router uses a round-robin algorithm to select the path that was not used for the last newly learned destination IP address. Once the router associates a path with a particular destination IP address, the router will always use that path as long as the router contains the destination IP address in its cache.

NOTE
The Layer 3 switch does not perform source routing. The router is concerned only with the paths to the next-hop routers, not the entire paths to the destination hosts.

A BGP4 destination can be learned from multiple BGP4 neighbors, leading to multiple BGP4 paths to reach the same destination. Each of the paths may be reachable through multiple IGP paths (multiple OSPF or RIP paths). In this case, the software installs all the multiple equal-cost paths in the BGP4 route table, up to the maximum number of BGP4 equal-cost paths allowed. The IP load sharing feature then distributes traffic across the equal-cost paths to the destination.

If an IGP path used by a BGP4 next-hop route path installed in the IP route table changes, then the BGP4 paths and IP paths are adjusted accordingly. For example, if one of the OSPF paths to reach the BGP4 next hop goes down, the software removes this path from the BGP4 route table and the IP route table. Similarly, if an additional OSPF path becomes available to reach the BGP4 next-hop router for a particular destination, the software adds the additional path to the BGP4 route table and the IP route table.

Changing the maximum number of shared BGP4 paths

When IP load sharing is enabled, BGP4 can balance traffic to a specific destination across up to four equal paths. You can set the maximum number of paths to a value from 1 through 4. The default is 1.

NOTE
The maximum number of BGP4 load sharing paths cannot be greater than the maximum number of IP load sharing paths. To increase the maximum number of IP load sharing paths, use the **ip load sharing** *num* command at the global CONFIG level of the CLI.

To change the maximum number of shared paths, enter commands such as the following.

```
Brocade(config)#router bgp
Brocade(config-bgp-router)#maximum-paths 4
Brocade(config-bgp-router)#write memory
```

Syntax: [**no**] **maximum-paths** *num*

The *num* parameter specifies the maximum number of paths across which the Layer 3 switch can balance traffic to a given BGP4 destination. You can change the maximum number of paths to a value from 2 through 4. The default is 1.

Customizing BGP4 load sharing

By default, when BGP4 load sharing is enabled, both IBGP and EBGP paths are eligible for load sharing, while paths from different neighboring autonomous systems are not eligible. You can change load sharing to apply only to IBGP or EBGP paths, or to support load sharing among paths from different neighboring autonomous systems.

To enable load sharing of IBGP paths only, enter the following command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#multipath ibgp
```

To enable load sharing of EBGP paths only, enter the following command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#multipath ebgp
```

To enable load sharing of paths from different neighboring autonomous systems, enter the following command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#multipath multi-as
```

Syntax: [**no**] **multipath ebgp** | **ibgp** | **multi-as**

The **ebgp** | **ibgp** | **multi-as** parameter specifies the change you are making to load sharing:

- **ebgp** – Load sharing applies only to EBGP paths. Load sharing is disabled for IBGP paths.
- **ibgp** – Load sharing applies only to IBGP paths. Load sharing is disabled for EBGP paths.
- **multi-as** – Load sharing is enabled for paths from different autonomous systems.

By default, load sharing applies to EBGP and IBGP paths, and does not apply to paths from different neighboring autonomous systems.

Specifying a list of networks to advertise

By default, the router sends BGP4 routes only for the networks you identify using the **network** command or that are redistributed into BGP4 from RIP or OSPF. You can specify up to 600 networks.

To specify a network to be advertised, use either of the following methods.

NOTE
The exact route must exist in the IP route table before the Layer 3 switch can create a local BGP route.

To configure the Layer 3 switch to advertise network 209.157.22.0/24, enter the following command.

```
Brocade(config-bgp-router)#network 192.168.22.0 255.255.255.0
```

Syntax: **network** *ip-addr ip-mask* [**nlri multicast** | **unicast** | **multicast unicast**] [**route-map** *map-name*] | [**weight** *num*] | [**backdoor**]

The *ip-addr* is the network number and the *ip-mask* specifies the network mask.

The **nlri multicast** | **unicast** | **multicast unicast** parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. Optionally, you also can specify **unicast** if you want the Layer 3 switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is **unicast** only.

The **route-map** *map-name* parameter specifies the name of the route map you want to use to set or change BGP4 attributes for the network you are advertising. The route map must already be configured.

The **weight** *num* parameter specifies a weight to be added to routes to this network.

The **backdoor** parameter changes the administrative distance of the route to this network from the EBGP administrative distance (20 by default) to the Local BGP weight (200 by default), thus tagging the route as a backdoor route. Use this parameter when you want the router to prefer IGP routes such as RIP or OSPF routes over the EBGP route for the network.

Specifying a route map name when configuring BGP4 network information

You can specify a route map as one of the parameters when you configure a BGP4 network to be advertised. The Layer 3 switch can use the route map to set or change BGP4 attributes when creating a local BGP4 route.

To configure network information and use a route map to set or change BGP4 attributes, use the following CLI method.

NOTE
You must configure the route map before you can specify the route map name in a BGP4 network configuration.

To configure a route map, and use it to set or change route attributes for a network you define for BGP4 to advertise, enter commands such as the following.

```
Brocade(config)#route-map set_net permit 1
Brocade(config-routemap set_net)#set community no-export
Brocade(config-routemap set_net)#exit
Brocade(config)#router bgp
Brocade(config-bgp-router)#network 10.100.1.0/24 route-map set_net
```

The first two commands in this example create a route map named “set_net” that sets the community attribute for routes that use the route map to “NO_EXPORT”. The next two commands change the CLI to the BGP4 configuration level. The last command configures a network for advertising from BGP4, and associates the “set_net” route map with the network. When BGP4 originates the 10.100.1.0/24 network, BGP4 also sets the community attribute for the network to “NO_EXPORT”.

Syntax: **network** *ip-addr ip-mask* [**route-map** *map-name*] | [**weight** *num*] | [**backdoor**]

The **route-map** *map-name* parameter specifies the name of the route map you want to use to set or change BGP4 attributes for the network you are advertising. The route map must already be configured.

For information about the other parameters, refer to "Defining route maps" on page 342.

Changing the default local preference

When the router uses the BGP4 algorithm to select a route to send to the IP route table, one of the parameters the algorithm uses is the local preference. Local preference is an attribute that indicates a degree of preference for a route relative to other routes. BGP4 neighbors can send the local preference value as an attribute of a route in an UPDATE message.

Local preference applies only to routes within the local AS. BGP4 routers can exchange local preference information with neighbors who also are in the local AS, but BGP4 routers do not exchange local preference information with neighbors in remote autonomous systems.

The default local preference is 100. For routes learned from EBGP neighbors, the default local preference is assigned to learned routes. For routes learned from IBGP neighbors, the local preference value is not changed for the route.

When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen.

NOTE
To set the local preference for individual routes, use route maps. Refer to "Defining route maps" on page 342. Refer to "How BGP4 selects a path for a route" on page 283 for information about the BGP4 algorithm.

To change the default local preference to 200, enter the following command.

```
Brocade(config-bgp-router)#default-local-preference 200
```

Syntax: default-local-preference *num*

The *num* parameter indicates the preference and can be a value from 0 through 4294967295.

Using the IP default route as a valid next hop for a BGP4 route

By default, the Layer 3 switch does not use a default route to resolve a BGP4 next-hop route. If the IP route lookup for the BGP4 next hop does not result in a valid IGP route (including static or direct routes), the BGP4 next hop is considered to be unreachable and the BGP4 route is not used.

In some cases, such as when the Layer 3 switch is acting as an edge router, you might want to allow the device to use the default route as a valid next hop. To do so, enter the following command at the BGP4 configuration level of the CLI.

```
Brocade(config-bgp-router)#next-hop-enable-default
```

Syntax: [**no**] **next-hop-enable-default**

Advertising the default route

By default, the Layer 3 switch does not originate and advertise a default route using BGP4. A BGP4 default route is the IP address 0.0.0.0 and the route prefix 0 or network mask 0.0.0.0. For example, 0.0.0.0/0 is a default route. You can enable the router to advertise a default BGP4 route using either of the following methods.

NOTE
The Brocade Layer 3 switch checks for the existence of an IGP route for 0.0.0.0/0 in the IP route table before creating a local BGP route for 0.0.0.0/0.

To enable the router to originate and advertise a default BGP4 route, enter the following command.

```
Brocade(config-bgp-router)#default-information-originate
```

Syntax: [**no**] **default-information-originate**

Changing the default MED (Metric) used for route redistribution

The Brocade Layer 3 switch can redistribute directly connected routes, static IP routes, RIP routes, and OSPF routes into BGP4. The MED (metric) is a global parameter that specifies the cost that will be applied to all routes by default when they are redistributed into BGP4. When routes are selected, lower metric values are preferred over higher metric values. The default BGP4 MED value is 0 and can be assigned a value from 0 through 4294967295.

NOTE
RIP and OSPF also have default metric parameters. The parameters are set independently for each protocol and have different ranges.

To change the default metric to 40, enter the following command.

```
Brocade(config-bgp-router)#default-metric 40
```

Syntax: **default-metric** *num*

The *num* indicates the metric and can be a value from 0 through 4294967295.

Enabling next-hop recursion

For each BGP4 route a Layer 3 switch learns, the Layer 3 switch performs a route lookup to obtain the IP address of the route next hop. A BGP4 route becomes eligible for installation into the IP route table only if the following conditions are true:

- The lookup succeeds in obtaining a valid next-hop IP address for the route.
- The path to the next-hop IP address is an Interior Gateway Protocol (IGP) path or a static route path.

By default, the software performs only one lookup for a BGP route next-hop IP address. If the next-hop lookup does not result in a valid next-hop IP address or the path to the next-hop IP address is a BGP path, the software considers the BGP route destination to be unreachable. The route is not eligible to be installed in the IP route table.

It is possible for the BGP route table to contain a route whose next-hop IP address is not reachable through an IGP route, even though a hop farther away can be reached by the Layer 3 switch through an IGP route. This can occur when the IGPs do not learn a complete set of IGP routes, resulting in the Layer 3 switch learning about an internal route through IBGP instead of through an IGP. In this case, the IP route table does not contain a route that can be used to reach the BGP route destination.

To enable the Layer 3 switch to find the IGP route to a BGP route next-hop gateway, enable recursive next-hop lookups. When you enable recursive next-hop lookup, if the first lookup for a BGP route results in an IBGP path originated within the same Autonomous System (AS), rather than an IGP path or static route path, the Layer 3 switch performs a lookup on the next-hop gateway next-hop IP address. If this second lookup results in an IGP path, the software considers the BGP route to be valid and thus eligible for installation in the IP route table. Otherwise, the Layer 3 switch performs a lookup on the next-hop IP address of the next-hop gateway next hop, and so on, until one of the lookups results in an IGP route.

NOTE

The software does not support using the default route to resolve a BGP4 route's next hop. Instead, you must configure a static route or use an IGP to learn the route to the EBGP multihop peer.

Previous software releases support use of the default route to resolve routes learned from EBGP multihop neighbors. However, even in this case Brocade recommends that you use a static route for the EBGP multihop neighbor instead. In general, we recommend that you do not use the default route as the next hop for BGP4 routes, especially when there are two or more BGP4 neighbors. Using the default route can cause loops.

Example when recursive route lookups are disabled

Here is an example of the results of an unsuccessful next-hop lookup for a BGP route. In this case, next-hop recursive lookups are disabled. The example is for the BGP route to network 192.168.0.0/24.

```
Brocade#show ip bgp route
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix             Next Hop        Metric     LocPrf      Weight Status
1      0.0.0.0/0           10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 80
2      192.168.0.0/24       10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 1
3      192.168.10.0/24       10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 1 189
4      192.168.50.0/24       10.0.0.1       1          100         0      I
         AS_PATH: 65001 4355 3356 7170 1455
5      192.168.19.0/24       10.157.24.1    1          100         0      I
         AS_PATH: 65001 4355 701
```

In this example, the Layer 3 switch cannot reach 192.168.0.0/24, because the next-hop IP address for the route is an IBGP route instead of an IGP route, and thus is considered unreachable by the Layer 3 switch. Here is the IP route table entry for the BGP route next-hop gateway (192.168.10.1/24).

```
Brocade#show ip route 10.0.0.1
Total number of IP routes: 37
      Network Address   NetMask           Gateway          Port    Cost   Type
      10.0.0.0         255.255.255.0     10.0.0.1         1/1/1     1      B
```

The route to the next-hop gateway is a BGP route, not an IGP route, and thus cannot be used to reach 192.168.0.0/24. In this case, the Layer 3 switch tries to use the default route, if present, to reach the subnet that contains the BGP route next-hop gateway.

```
Brocade#show ip route 240.0.0.0/24
Total number of IP routes: 37
      Network Address   NetMask           Gateway          Port    Cost   Type
      0.0.0.0           0.0.0.0           10.0.0.202       1/1/1     1      S
```

Example when recursive route lookups are enabled

When recursive next-hop lookups are enabled, the Layer 3 switch recursively looks up the next-hop gateways along the route until the Layer 3 switch finds an IGP route to the BGP route destination. Here is an example.

```
Brocade#show ip bgp route
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix         Next Hop        Metric     LocPrf      Weight Status
1      0.0.0.0/0       10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 80
2      192.168.0.0/24    10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 1
3      192.168.10.0/24    10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 1 189
4      192.168.30.0/24   10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 3356 7170 1455
5      192.160.80.0/24   10.157.24.1    1          100         0       I
         AS_PATH: 65001 4355 701
```

The first lookup results in an IBGP route, to network 192.168.0.0/24.

```
Brocade#show ip route 192.168.0.1
Total number of IP routes: 38
      Network Address   NetMask           Gateway           Port    Cost   Type
      192.168.0.0         255.255.255.0     10.0.0.1          1/1/1     1      B
         AS_PATH: 65001 4355 1
```

Since the route to 192.168.0.1/24 is not an IGP route, the Layer 3 switch cannot reach the next hop through IP, and thus cannot use the BGP route. In this case, since recursive next-hop lookups are enabled, the Layer 3 switch next performs a lookup for 192.168.0.1 next-hop gateway, 10.0.0.1.

```
Brocade#show ip bgp route 192.168.0.0
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix         Next Hop        Metric     LocPrf      Weight Status
1      192.168.0.0/24   10.0.0.1        1          100        0      BI
         AS_PATH: 65001 4355 1
```

The next-hop IP address for 192.0.0.1 is not an IGP route, which means the BGP route destination still cannot be reached through IP. The recursive next-hop lookup feature performs a lookup on 10.0.0.1 next-hop gateway.

```
Brocade#show ip route 10.0.0.1
Total number of IP routes: 38
      Network Address   NetMask           Gateway            Port    Cost   Type
      10.0.0.0         255.255.255.0     0.0.0.0           1/1/1     1      D
        AS_PATH: 65001 4355 1
```

This lookup results in an IGP route. In fact, this route is a directly-connected route. As a result, the BGP route destination is now reachable through IGP, which means the BGP route is eligible for installation in the IP route table. Here is the BGP route in the IP route table.

```
Brocade#show ip route 192.168.0.0/24
Total number of IP routes: 38
      Network Address    NetMask          Gateway            Port    Cost   Type
      192.168.0.0          255.255.255.0    10.0.0.1           1/1/1     1      B
         AS_PATH: 65001 4355 1
```

This Layer 3 switch can use this route because the Layer 3 switch has an IP route to the next-hop gateway. Without recursive next-hop lookups, this route would not be in the IP route table.

Enabling recursive next-hop lookups

The recursive next-hop lookups feature is disabled by default. To enable recursive next-hop lookups, enter the **next-hop-recursion** command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#next-hop-recursion
```

Syntax: [no] next-hop-recursion

Changing administrative distances

BGP4 routers can learn about networks from various protocols, including the EBGP portion of BGP4 and IGPs such as OSPF and RIP. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned.

To select one route over another based on the source of the route information, the Layer 3 switch can use the administrative distances assigned to the sources. The administrative distance is a protocol-independent metric that IP routers use to compare routes from different sources.

The Layer 3 switch re-advertises a learned best BGP4 route to the Layer 3 switch neighbors even when the software does not also select that route for installation in the IP route table. The best BGP4 routes is the BGP4 path that the software selects based on comparison of the paths' BGP4 route parameters. Refer to "How BGP4 selects a path for a route" on page 283.

When selecting a route from among different sources (BGP4, OSPF, RIP, static routes, and so on), the software compares the routes on the basis of each route administrative distance. If the administrative distance of the paths is lower than the administrative distance of paths from other sources (such as static IP routes, RIP, or OSPF), the BGP4 paths are installed in the IP route table.

NOTE
The software will replace a statically configured default route with a learned default route if the learned route administrative distance is lower than the statically configured default route distance. However, the default administrative distance for static routes is changed to 1, so only directly-connected routes are preferred over static routes when the default administrative distances for the routes are used.

The following default administrative distances are found on the Brocade Layer 3 switch:

- Directly connected – 0 (this value is not configurable)
- Static – 1 (applies to all static routes, including default routes)
- EBGP – 20
- OSPF – 110
- RIP – 120
- IBGP – 200
- Local BGP – 200
- Unknown – 255 (the router will not use this route)

Lower administrative distances are preferred over higher distances. For example, if the router receives routes for the same network from OSPF and from RIP, the router will prefer the OSPF route by default. The administrative distances are configured in different places in the software. The Layer 3 switch re-advertises a learned best BGP4 route to neighbors by default, regardless of whether the route administrative distance is lower than other routes from different route sources to the same destination.

- To change the EBGP, IBGP, and Local BGP default administrative distances, see the instructions in this section.
- To change the default administrative distance for OSPF, refer to “Administrative distance” on page 207.
- To change the default administrative distance for RIP, refer to “Changing the administrative distance” on page 146.
- To change the default administrative distance for static routes, refer to “Static routes configuration” on page 45.

You can change the default EBGP, IBGP, and Local BGP administrative distances using either of the following methods.

To change the default administrative distances for EBGP, IBGP, and Local BGP, enter a command such as the following.

```
Brocade(config-bgp-router)#distance 180 160 40
```

Syntax: **distance** *external-distance internal-distance local-distance*

The *external-distance* sets the EBGP distance and can be a value from 1 through 255.

The *internal-distance* sets the IBGP distance and can be a value from 1 through 255.

The *local-distance* sets the Local BGP distance and can be a value from 1 through 255.

Requiring the first AS to be the neighbor AS

By default, the Brocade device does not require the first AS listed in the AS_SEQUENCE field of an AS path Update from an EBGP neighbor to be the AS that the neighbor who sent the Update is in. You can enable the Brocade device for this requirement.

When you enable the Brocade device to require the AS that an EBGP neighbor is in to be the same as the first AS in the AS_SEQUENCE field of an Update from the neighbor, the Brocade device accepts the Update only if the autonomous systems match. If the autonomous systems do not match, the Brocade device sends a Notification message to the neighbor and closes the session. The requirement applies to all Updates received from EBGP neighbors.

To enable this feature, enter the following command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#enforce-first-as
```

Syntax: [no] **enforce-first-as**

Disabling or re-enabling comparison of the AS-Path length

AS-Path comparison is Step 5 in the algorithm BGP4 uses to select the next path for a route. Comparison of the AS-Path length is enabled by default. To disable it, enter the following command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#as-path-ignore
```

This command disables comparison of the AS-Path lengths of otherwise equal paths. When you disable AS-Path length comparison, the BGP4 algorithm shown in "How BGP4 selects a path for a route" on page 283 skips from Step 4 to Step 6.

Syntax: [no] **as-path-ignore**

Enabling or disabling comparison of the router IDs

Router ID comparison is Step 10 in the algorithm BGP4 uses to select the next path for a route.

NOTE
Comparison of router IDs is applicable only when BGP4 load sharing is disabled.

When router ID comparison is enabled, the path comparison algorithm compares the router IDs of the neighbors that sent the otherwise equal paths:

- If BGP4 load sharing is disabled (maximum-paths 1), the Layer 3 switch selects the path that came from the neighbor with the lower router ID.
- If BGP4 load sharing is enabled, the Layer 3 switch load shares among the remaining paths. In this case, the router ID is not used to select a path.

NOTE
Router ID comparison is disabled by default. In previous releases, router ID comparison is enabled by default and cannot be disabled.

To enable router ID comparison, enter the **compare-routerid** command at the BGP configuration level of the CLI.

```
Brocade(config-bgp-router)#compare-routerid
```

Syntax: [no] **compare-routerid**

For more information, refer to "How BGP4 selects a path for a route" on page 283.

Configuring the Layer 3 switch to always compare Multi-Exit Discriminators

A Multi-Exit Discriminator (MED) is a value that the BGP4 algorithm uses when comparing multiple paths received from different BGP4 neighbors in the same AS for the same route. In BGP4, a route MED is equivalent to its "metric":

- BGP4 compares the MEDs of two otherwise equivalent paths ***if and only if*** the routes were learned from the same neighboring AS. This behavior is called ***deterministic MED***. Deterministic MED is always enabled and cannot be disabled.

  In addition, you can enable the Layer 3 switch to always compare the MEDs, regardless of the AS information in the paths. To enable this comparison, enter the **always-compare-med** command at the BGP4 configuration level of the CLI. This option is disabled by default.

- The Layer 3 switch compares the MEDs based on one or more of the following conditions. By default, the Layer 3 switch compares the MEDs of paths ***only if*** the first AS in the paths is the same. (The Layer 3 switch skips over the AS-CONFED-SEQUENCE if present.)

You can enable the Layer 3 switch to always compare the MEDs, regardless of the AS information in the paths. For example, if the router receives UPDATES for the same route from neighbors in three autonomous systems, the router would compare the MEDs of all the paths together, rather than comparing the MEDs for the paths in each AS individually.

NOTE
By default, value 0 (most favorable) is used in MED comparison when the MED attribute is not present. The default MED comparison results in the Layer 3 switch favoring the route paths that are missing their MEDs. You can use the **med-missing-as-worst** command to make the Layer 3 switch regard a BGP route with a missing MED attribute as the least favorable route, when comparing the MEDs of the routes.

NOTE
MED comparison is not performed for internal routes originated within the local AS or confederation.

To configure the router to always compare MEDs, enter the following command.

```
Brocade(config-bgp-router)#always-compare-med
```

Syntax: [no] **always-compare-med**

Treating missing MEDs as the worst MEDs

By default, the Layer 3 switch favors a lower MED over a higher MED during MED comparison. Since the Layer 3 switch assigns the value 0 to a route path MED if the MED value is missing, the default MED comparison results in the Layer 3 switch favoring the route paths that are missing their MEDs.

To change this behavior so that the Layer 3 switch favors a route that has a MED over a route that is missing its MED, enter the following command at the BGP4 configuration level of the CLI.

```
Brocade(config-bgp-router)#med-missing-as-worst
```

Syntax: [no] **med-missing-as-worst**

NOTE
This command affects route selection only when route paths are selected based on MED comparison. It is still possible for a route path that is missing its MED to be selected based on other criteria. For example, a route path with no MED can be selected if its weight is larger than the weights of the other route paths.

Route reflection parameter configuration

Normally, all the BGP routers within an AS are fully meshed. Each of the routers has an IBGP session with each of the other BGP routers in the AS. Each IBGP router thus has a route for each of its IBGP neighbors. For large autonomous systems containing many IBGP routers, the IBGP route information in each of the fully-meshed IBGP routers can introduce too much administrative overhead.

To avoid this problem, you can hierarchically organize your IGP routers into clusters:

- A **cluster** is a group of IGP routers organized into route reflectors and route reflector clients. You configure the cluster by assigning a cluster ID on the route reflector and identifying the IGP neighbors that are members of that cluster. All the configuration for route reflection takes place on the route reflectors. The clients are unaware that they are members of a route reflection cluster. All members of the cluster must be in the same AS. The cluster ID can be any number from 0 through 4294967295. The default is the router ID, expressed as a 32-bit number.

  NOTE
  If the cluster contains more than one route reflector, you need to configure the same cluster ID on all the route reflectors in the cluster. The cluster ID helps route reflectors avoid loops within the cluster.

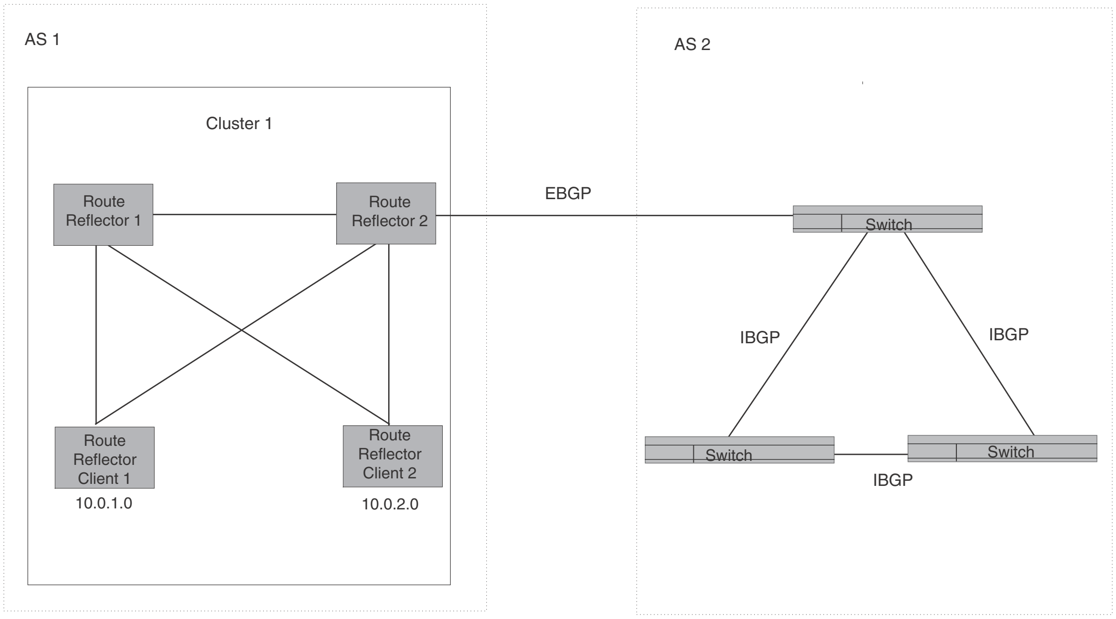
- A **route reflector** is an IGP router configured to send BGP route information to all the clients (other BGP4 routers) within the cluster. Route reflection is enabled on all Brocade BGP4 routers by default but does not take effect unless you add route reflector clients to the router.
- A **route reflector client** is an IGP router identified as a member of a cluster. You identify a router as a route reflector client on the router that is the route reflector, not on the client. The client itself requires no additional configuration. In fact, the client does not know that it is a route reflector client. The client just knows that it receives updates from its neighbors and does not know whether one or more of those neighbors are route reflectors.

NOTE
Route reflection applies only among IBGP routers within the same AS. You cannot configure a cluster that spans multiple autonomous systems.

Figure 26 shows an example of a route reflector configuration. In this example, two Layer 3 switches are configured as route reflectors for the same cluster. The route reflectors provide redundancy in case one of the reflectors becomes unavailable. Without redundancy, if a route reflector becomes unavailable, its clients are cut off from BGP4 updates.

AS1 contains a cluster with two route reflectors and two clients. The route reflectors are fully meshed with other BGP4 routers, but the clients are not fully meshed. They rely on the route reflectors to propagate BGP4 route updates.

FIGURE 26 Example of a route reflector configuration



Route reflection based on RFC 2796

Route reflection on Brocade devices is based on RFC 2796. This updated RFC helps eliminate routing loops that are possible in some implementations of the older specification, RFC 1966.

NOTE
The configuration procedure for route reflection is the same regardless of whether your software release is using RFC 1966 or RFC 2796. However, the operation of the feature is different as explained below.

RFC 2796 provides more details than RFC 1966 regarding the use of the route reflection attributes, ORIGINATOR_ID and CLUSTER_LIST, to help prevent loops:

- ORIGINATOR_ID – Specifies the router ID of the BGP4 switch that originated the route. The route reflector inserts this attribute when reflecting a route to an IBGP neighbor. If a BGP4 switch receives an advertisement that contains its own router ID as the ORIGINATOR_ID, the switch discards the advertisement and does not forward it.
- CLUSTER_LIST – A list of the route reflection clusters through which the advertisement has passed. A cluster contains a route reflector and its clients. When a route reflector reflects a route, the route reflector adds its cluster ID to the front of the CLUSTER_LIST. If a route reflector receives a route that has its own cluster ID, the switch discards the advertisement and does not forward it.

The Brocade device handles the attributes as follows:

- The Layer 3 switch adds the attributes only if it is a route reflector, and only when advertising IBGP route information to other IBGP neighbors. The attributes are not used when communicating with EBGP neighbors.
- A Layer 3 switch configured as a route reflector sets the ORIGINATOR_ID attribute to the router ID of the router that originated the route. Moreover, the route reflector sets the attribute only if this is the first time the route is being reflected (sent by a route reflector). In previous software releases, the route reflector set the attribute to the router ID of the route reflector itself. When a Layer 3 switch receives a route that already has the ORIGINATOR_ID attribute set, the Layer 3 switch does not change the value of the attribute.
- If a Layer 3 switch receives a route whose ORIGINATOR_ID attribute has the value of the Layer 3 switch own router ID, the Layer 3 switch discards the route and does not advertise it. By discarding the route, the Layer 3 switch prevents a routing loop. The Layer 3 switch did not discard the route in previous software releases.
- The first time a route is reflected by a Layer 3 switch configured as a route reflector, the route reflector adds the CLUSTER_LIST attribute to the route. Other route reflectors who receive the route from an IBGP neighbor add their cluster IDs to the front of the route CLUSTER_LIST. If the route reflector does not have a cluster ID configured, the Layer 3 switch adds its router ID to the front of the CLUSTER_LIST.
- If a Layer 3 switch configured as a route reflector receives a route whose CLUSTER_LIST contains the route reflector own cluster ID, the route reflector discards the route and does not forward it.

Configuration procedures for BGP4 route reflector

To configure a Brocade Layer 3 switch to be a BGP4 route reflector, use either of the following methods.

NOTE
All configuration for route reflection takes place on the route reflectors, not on the clients.

Enter the following commands to configure a Brocade Layer 3 switch as route reflector 1 in Figure 26 on page 318. To configure route reflector 2, enter the same commands on the Layer 3 switch that will be route reflector 2. The clients require no configuration for route reflection.

```
Brocade(config-bgp-router)#cluster-id 1
Brocade(config-bgp-router)#neighbor 10.0.1.0 route-reflector-client
Brocade(config-bgp-router)#neighbor 10.0.2.0 route-reflector-client
```

Syntax: [**no**] **cluster-id** *num* | *ip-addr*

The *num* | *ip-addr* parameter specifies the cluster ID and can be a number from 0 through 4294967295 or an IP address. The default is the router ID. You can configure one cluster ID on the router. All route-reflector clients for the router are members of the cluster.

NOTE
If the cluster contains more than one route reflector, you need to configure the same cluster ID on all the route reflectors in the cluster. The cluster ID helps route reflectors avoid loops within the cluster.

To add an IBGP neighbor to the cluster, enter the following command.

Syntax: **neighbor** *ip-addr* **route-reflector-client**

For more information about the **neighbor** command, refer to "Adding BGP4 neighbors" on page 292.

By default, the clients of a route reflector are not required to be fully meshed; the routes from a client are reflected to other clients. However, if the clients are fully meshed, route reflection is not required between clients.

If you need to disable route reflection between clients, enter the following command. When the feature is disabled, route reflection does not occur between clients but reflection does still occur between clients and non-clients.

```
Brocade(config-bgp-router)#no client-to-client-reflection
```

Enter the following command to re-enable the feature.

```
Brocade(config-bgp-router)#client-to-client-reflection
```

Syntax: [**no**] **client-to-client-reflection**

Configuration notes for BGP4 autonomous systems

A **confederation** is a BGP4 Autonomous System (AS) that has been subdivided into multiple, smaller autonomous systems. Subdividing an AS into smaller autonomous systems simplifies administration and reduces BGP-related traffic, thus reducing the complexity of the Interior Border Gateway Protocol (IBGP) mesh among the BGP routers in the AS.

The Brocade implementation of this feature is based on RFC 3065.

Normally, all BGP routers within an AS must be fully meshed, so that each BGP router has interfaces to all the other BGP routers within the AS. This is feasible in smaller autonomous systems but becomes unmanageable in autonomous systems containing many BGP routers.

When you configure BGP routers into a confederation, all the routers within a sub-AS (a subdivision of the AS) use IBGP and must be fully meshed. However, routers use EBGP to communicate between different sub-autonomous systems.

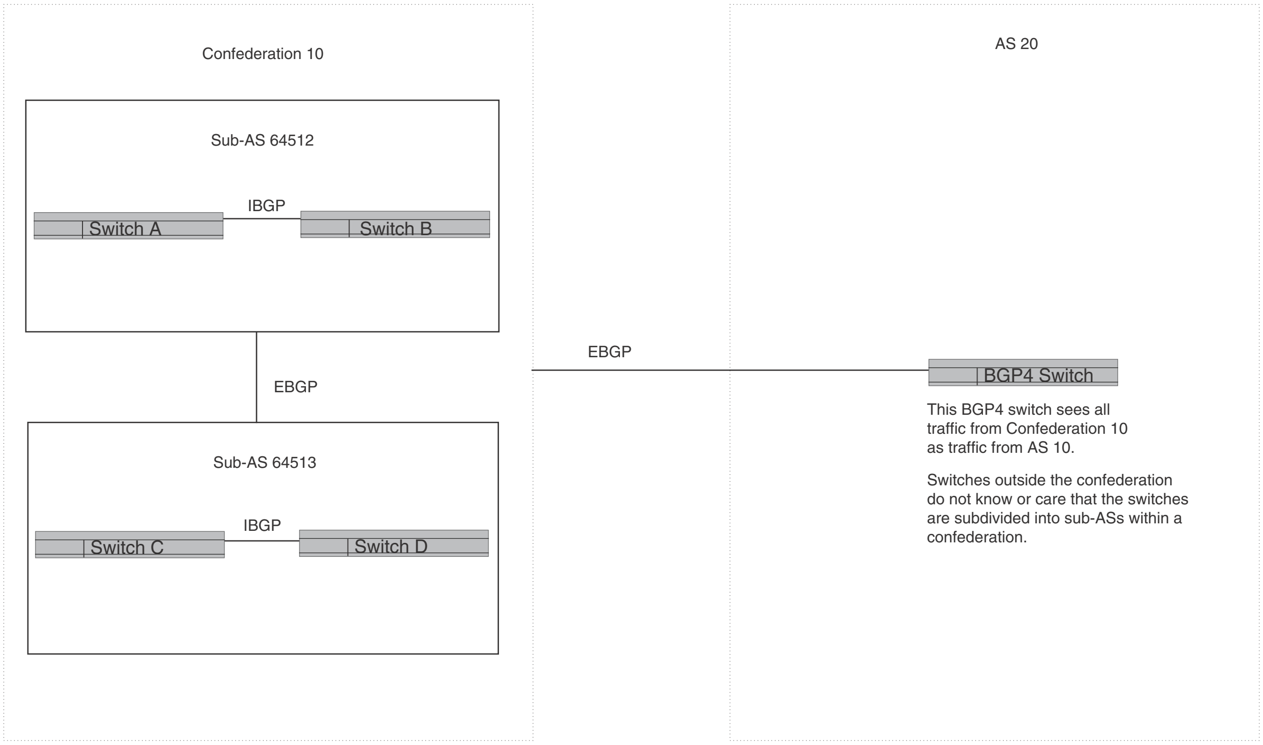
NOTE
Another method for reducing the complexity of an IBGP mesh is to use route reflection. However, if you want to run different Interior Gateway Protocols (IGPs) within an AS, configure a confederation. You can run a separate IGP within each sub-AS.

To configure a confederation, configure groups of BGP routers into sub-autonomous systems. A sub-AS is simply an AS. The term "sub-AS" distinguishes autonomous systems within a confederation from autonomous systems that are not in a confederation. For the viewpoint of remote autonomous systems, the confederation ID is the AS ID. Remote autonomous systems do not know that the AS represents multiple sub-autonomous systems with unique AS IDs.

NOTE
You can use any valid AS numbers for the sub-autonomous systems. If your AS is connected to the Internet, Brocade recommends that you use numbers from within the private AS range (64512 through 65535). These are private autonomous systems numbers and BGP4 routers do not propagate these AS numbers to the Internet.

Figure 27 shows an example of a BGP4 confederation.

FIGURE 27 **Example of a BGP4 confederation**


Confederation 10
Sub-AS 64512
IBGP
Switch A
Switch B
EBGP
Sub-AS 64513
IBGP
Switch C
Switch D
EBGP
AS 20
BGP4 Switch
This BGP4 switch sees all traffic from Confederation 10 as traffic from AS 10.
Switches outside the confederation do not know or care that the switches are subdivided into sub-ASs within a confederation.

In this example, four switches are configured into two sub-autonomous systems, each containing two of the switches. The sub-autonomous systems are members of confederation 10. Switches within a sub-AS must be fully meshed and communicate using IBGP. In this example, Switches A and B use IBGP to communicate. Switches C and D also use IBGP. However, the sub-autonomous systems communicate with one another using EBGP. For example, Switch A communicates with Switch C using EBGP. The switches in the confederation communicate with other autonomous systems using EBGP.

Switches in other autonomous systems are unaware that Switches A through D are configured in a confederation. In fact, when switches in confederation 10 send traffic to switches in other autonomous systems, the confederation ID is the same as the AS number for the switches in the confederation. Thus, switches in other autonomous systems see traffic from AS 10 and are unaware that the switches in AS 10 are subdivided into sub-autonomous systems within a confederation.

Configuring a BGP confederation

Perform the following configuration tasks on each BGP router within the confederation:

- Configure the local AS number. The local AS number indicates membership in a sub-AS. All BGP switches with the same local AS number are members of the same sub-AS. BGP switches use the local AS number when communicating with other BGP switches within the confederation.

- Configure the confederation ID. The confederation ID is the AS number by which BGP switches outside the confederation know the confederation. Thus, a BGP switch outside the confederation is not aware and does not care that your BGP switches are in multiple sub-autonomous systems. BGP switches use the confederation ID when communicating with switches outside the confederation. The confederation ID must be different from the sub-AS numbers.
- Configure the list of the sub-AS numbers that are members of the confederation. All the switches within the same sub-AS use IBGP to exchange switch information. Switches in different sub-autonomous systems within the confederation use EBGP to exchange switch information.

To configure four Layer 3 switches to be a member of confederation 10 (as shown in Figure 27), consisting of two sub-autonomous systems (64512 and 64513), enter commands such as the following.

Commands for router A

```
BrocadeA(config)#router bgp
BrocadeA(config-bgp-router)#local-as 64512
BrocadeA(config-bgp-router)#confederation identifier 10
BrocadeA(config-bgp-router)#confederation peers 64512 64513
BrocadeA(config-bgp-router)#write memory
```

Syntax: **local-as** *num*

The *num* parameter with the **local-as** command indicates the AS number for the BGP switches within the sub-AS. You can specify a number from 1 through 65535. Brocade recommends that you use a number within the range of well-known private autonomous systems, 64512 through 65535.

Syntax: **confederation identifier** *num*

The *num* parameter with the **confederation identifier** command indicates the confederation number. The confederation ID is the AS number by which BGP switches outside the confederation know the confederation. Thus, a BGP switch outside the confederation is not aware and does not care that your BGP switches are in multiple sub-autonomous systems. BGP switches use the confederation ID when communicating with switches outside the confederation. The confederation ID must be different from the sub-AS numbers. You can specify a number from 1 through 65535.

Syntax: **confederation peers** *num [num* ...]

The *num* parameter with the **confederation peers** command indicates the sub-AS numbers for the sub-autonomous systems in the confederation. You must specify all the sub-autonomous systems contained in the confederation. All the switches within the same sub-AS use IBGP to exchange switch information. Switches in different sub-autonomous systems within the confederation use EBGP to exchange switch information. You can specify a number from 1 through 65535.

Commands for router B

```
BrocadeB(config)#router bgp
BrocadeB(config-bgp-router)#local-as 64512
BrocadeB(config-bgp-router)#confederation identifier 10
BrocadeB(config-bgp-router)#confederation peers 64512 64513
BrocadeB(config-bgp-router)#write memory
```

Commands for router C

```
BrocadeC(config)#router bgp
BrocadeC(config-bgp-router)#local-as 64513
BrocadeC(config-bgp-router)#confederation identifier 10
BrocadeC(config-bgp-router)#confederation peers 64512 64513
BrocadeC(config-bgp-router)#write memory
```

Commands for router D

```
BrocadeD(config)#router bgp
BrocadeD(config-bgp-router)#local-as 64513
BrocadeD(config-bgp-router)#confederation identifier 10
BrocadeD(config-bgp-router)#confederation peers 64512 64513
BrocadeD(config-bgp-router)#write memory
```

Aggregating routes advertised to BGP4 neighbors

By default, the Layer 3 switch advertises individual routes for all the networks. The aggregation feature allows you to configure the Layer 3 switch to aggregate routes in a range of networks into a single CIDR number. For example, without aggregation, the Layer 3 switch will individually advertise routes for networks 10.95.1.0, 10.95.2.0, and 10.95.3.0. You can configure the Layer 3 switch to instead send a single, aggregate route for the networks. The aggregate route would be advertised as 10.95.0.0.

NOTE
To summarize CIDR networks, you must use the aggregation feature. The auto summary feature does not summarize networks that use CIDR numbers instead of class A, B, or C numbers.

To aggregate routes for 10.157.22.0, 10.157.23.0, and 10.157.24.0, enter the following command.

```
Brocade(config-bgp-router)#aggregate-address 10.157.0.0 255.255.0.0
```

Syntax: **aggregate-address** *ip-addr ip-mask* [**as-set**] [**nlri multicast** | **unicast** | **multicast unicast**] [**summary-only**] [**suppress-map** *map-name*] [**advertise-map** *map-name*] [**attribute-map** *map-name*]

The *ip-addr* and *ip-mask* parameters specify the aggregate value for the networks. Specify 0 for the host portion and for the network portion that differs among the networks in the aggregate. For example, to aggregate 10.0.1.0, 10.0.2.0, and 10.0.3.0, enter the IP address 10.0.0.0 and the network mask 255.255.0.0.

The **as-set** parameter causes the router to aggregate AS-path information for all the routes in the aggregate address into a single AS-path.

The **nlri multicast** | **unicast** | **multicast unicast** parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. Optionally, you also can specify **unicast** if you want the Layer 3 switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is **unicast** only.

The **summary-only** parameter prevents the router from advertising more specific routes contained within the aggregate route.

The **suppress-map** *map-name* parameter prevents the more specific routes contained in the specified route map from being advertised.

The **advertise-map** *map-name* parameter configures the router to advertise the more specific routes in the specified route map.

The **attribute-map** *map-name* parameter configures the router to set attributes for the aggregate routes based on the specified route map.

NOTE
For the **suppress-map**, **advertise-map**, and **attribute-map** parameters, the route map must already be defined. Refer to "Defining route maps" on page 342 for information on defining a route map.

Configuring BGP4 graceful restart

By default, BGP4 graceful restart is enabled for the global routing instance. This section describes how to disable and re-enable the BGP4 restart feature and change the default values for associated timers.

For information about displaying BGP4 graceful restart neighbor information, refer to "Displaying BGP4 graceful restart neighbor information" on page 390.

BGP4 graceful restart is enabled by default on a Layer 3 switch. To disable it, use the following commands:

```
Brocade(config)# router bgp
Brocade(config-bgp)# no graceful-restart
```

To re-enable BGP4 graceful restart after it has been disabled, enter the following commands.

```
Brocade(config)# router bgp
Brocade(config-bgp)# graceful-restart
```

Syntax: [no] **graceful-restart**

Configuring timers for BGP4 graceful restart (optional)

You can change the default values for the following timers:

- Restart timer
- Stale routes timer
- Purge timer

Configuring the restart timer for BGP4 graceful restart

Use the following command to specify the maximum amount of time a device will maintain routes from and forward traffic to a restarting device.

```
Brocade(config-bgp)# graceful-restart restart-timer 150
```

Syntax: [no] **graceful-restart restart-timer** *seconds*

The *seconds* variable is the maximum restart wait time advertised to neighbors. Possible values are from 1 through 3600 seconds. The default value is 120 seconds.

Configuring the BGP4 graceful restart stale routes timer

Use the following command to specify the maximum amount of time a helper device will wait for an end-of-RIB message from a peer before deleting routes from that peer.

```
Brocade(config-bgp)# graceful-restart stale-routes-time 120
```

Syntax: [no] **graceful-restart stale-routes-time** *seconds*

The *seconds* variable is the maximum time before a helper device cleans up stale routes. Possible values are from 1 through 3600 seconds. The default value is 360 seconds.

Configuring the BGP4 graceful restart purge timer

Use the following command to specify the maximum amount of time a device will maintain stale routes in its routing table before purging them.

```
Brocade(config-bgp)# graceful-restart purge-time 900
```

Syntax: [no] **graceful-restart purge-time** *seconds*

The seconds variable sets the maximum time before a restarting device cleans up stale routes. Possible values are from 1 through 3600 seconds. The default value is 600 seconds.

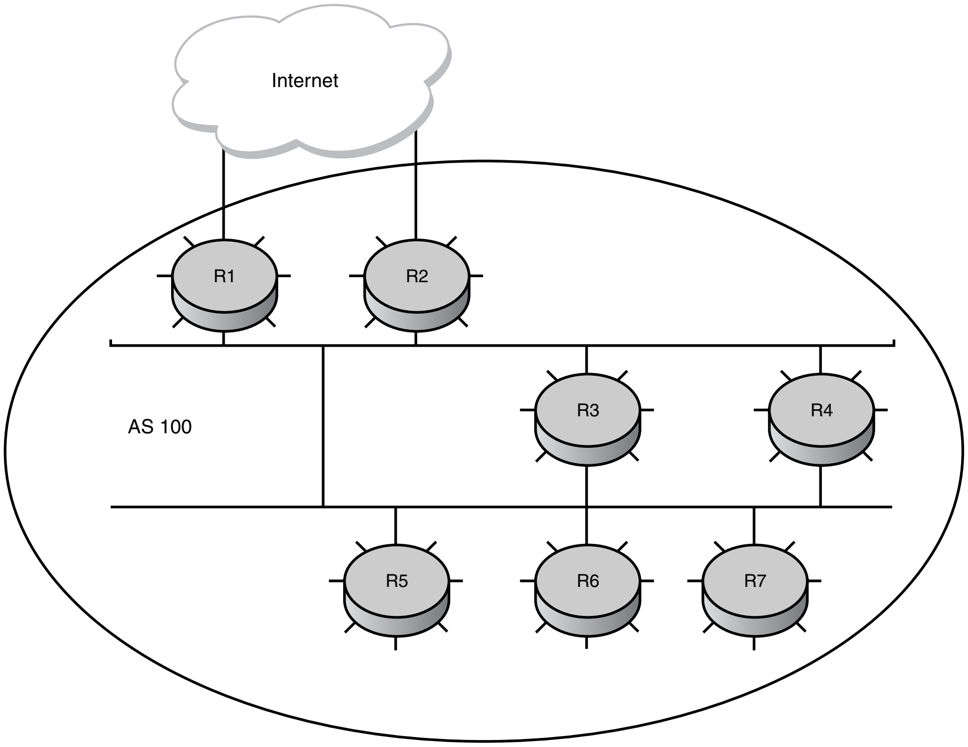
BGP null0 routing

The null0 routes were previously treated as invalid routes for BGP next hop resolution. BGP now uses the null0 route to resolve its next hop. Thus, null0 route in the routing table (for example, static route) is considered as a valid route by BGP. If the next hop for BGP resolves into a null0 route, the BGP route is also installed as a null0 route in the routing table.

The null0 routing feature allows network administrators to block certain network prefixes, by using null0 routes and route-maps. The combined use of null0 routes and route maps blocks traffic from a particular network prefix, telling a remote router to drop all traffic for this network prefix by redistributing a null0 route into BGP.

Figure 28 shows a topology for a null0 routing application example.

FIGURE 28 Example of a null0 routing application



The following steps configure a null0 routing application for stopping denial of service attacks from remote hosts on the internet.

Configuration steps for BGP null0 routing

1. Select one switch, S6, to distribute null0 routes throughout the BGP network.
2. Configure a route-map to match a particular tag (50) and set the next-hop address to an unused network address (199.199.1.1).
3. Set the local-preference to a value higher than any possible internal or external local-preference (50).
4. Complete the route map by setting origin to IGP.
5. On S6, redistribute the static routes into BGP, using route-map *route-map-name* (redistribute static route-map block user).
6. On S1, the router facing the internet, configure a null0 route matching the next-hop address in the route-map (ip route 199.199.1.1/32 null0).
7. Repeat step 3 for all switches interfacing with the internet (edge corporate routers). In this case, S2 has the same null0 route as S1.
8. On S6, configure the network prefixes associated with the traffic you want to drop. The static route IP address references a destination address. You are required to point the static route to the egress port, for example, Ethernet 1/1/2, and specify the tag 50, matching the route-map configuration.

Configuration examples for BGP null0 routing

S6

The following configuration defines specific prefixes to filter.

```
Brocade(config)#ip route 10.0.0.40/29 ethernet 1/1/2 tag 50
Brocade(config)#ip route 10.0.0.192/27 ethernet 1/1/2 tag 50
Brocade(config)#ip route 10.0.14.0/23 ethernet 1/1/2 tag 50
```

The following configuration redistributes routes into BGP.

```
Brocade(config)#router bgp
Brocade(config-bgp-router)#local-as 100
Brocade(config-bgp-router)#neighbor <router1_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router2_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router3_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router4_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router5_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router7_int_ip address> remote-as 100
Brocade(config-bgp-router)#redistribute static route-map blockuser
Brocade(config-bgp-router)#exit
```

The following configuration defines the specific next hop address and sets the local preference to preferred.

```
Brocade(config)#route-map blockuser permit 10
Brocade(config-routemap blockuser)#match tag 50
Brocade(config-routemap blockuser)#set ip next-hop 199.199.1.1
Brocade(config-routemap blockuser)#set local-preference 1000000
Brocade(config-routemap blockuser)#set origin igp
Brocade(config-routemap blockuser)#exit
```

S1

The following configuration defines the null0 route to the specific next hop address. The next hop address 199.199.1.1 points to the null0 route.

```
Brocade(config)#ip route 199.199.1.1/32 null0
Brocade(config)#router bgp
Brocade(config-bgp-router)#local-as 100
Brocade(config-bgp-router)#neighbor <router2_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router3_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router4_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router5_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router6_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router7_int_ip address> remote-as 100
```

S2

The following configuration defines a null0 route to the specific next hop address. The next hop address 199.199.1.1 points to the null0 route, which gets blocked.

```
Brocade(config)#ip route 199.199.1.1/32 null0
Brocade(config)#router bgp
Brocade(config-bgp-router)#local-as 100
Brocade(config-bgp-router)#neighbor <router1_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router3_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router4_int_ip address> remote-as 100
Brocade (config-bgp-router)#neighbor <router5_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router6_int_ip address> remote-as 100
Brocade(config-bgp-router)#neighbor <router7_int_ip address> remote-as 100
```

Show commands for BGP null0 routing

After configuring the null0 application, you can display the output.

S6

The following is the **show ip route static** output for S6.

```
Brocade#show ip route static
Type Codes - B:BGP  D:Connected  S:Static  R:RIP  O:OSPF;   Cost - Dist/Metric
          Destination          Gateway         Port           Cost     Type
1       10.0.0.40/29          DIRECT          eth 1/1/2     1/1       S
2       10.0.0.192/27         DIRECT          eth 1/1/2     1/1       S
3       10.0.14.0/23          DIRECT          eth 1/1/2     1/1       S
```

S1 and S2

The following is the **show ip route static** output for S1 and S2.

```
Brocade#show ip route static
 Type Codes - B:BGP  D:Connected  S:Static  R:RIP  O:OSPF;   Cost - Dist/Metric
             Destination        Gateway         Port        Cost      Type
1            199.199.1.1/32     DIRECT          drop        1/1/1       S
```

S6

The following is the **show ip bgp route** output for S6

```
Brocade#show ip bgp route
Total number of BGP Routes: 126
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED  E:EBGP
H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED s:STALE
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.1.0/24        10.0.1.3        0          100        0      BI
         AS_PATH:
.           ..                       .                       .           .
.
9      10.0.0.16/30      10.10.1.3                   100        0      I
         AS_PATH: 85
10     10.0.0.40/29      199.199.1.1/32  1        1000000 32768  BL
         AS_PATH:
11     10.0.0.80/28      10.10.1.3                   100        0      I
 .          ..                     .                       .          .       .
 .          ..                     .                       .          .
.
36     10.0.0.96/28      10.0.1.3                   100        0      I
         AS_PATH: 50
37     10.0.0.192/27     199.199.1.1/32  1       10000000 32768  BL
         AS_PATH:
 .             ..                  .                         .          .
.
64     10.0.7.0/24       10.20.1.3                   100        0      I
         AS_PATH: 10
65     10.0.14.0/23      199.199.1.1/32  1        1000000 32768  BL
        AS_PATH: ..                 .                      .            .     .
```

S1 and S2

The **show ip route** output for S1 and S2 shows "drop" under the Port column for the network prefixes you configured with null0 routing.

```
Brocade#show ip route
Total number of IP routes: 133
 Type Codes - B:BGP  D:Connected  S:Static  R:RIP  O:OSPF;   Cost - Dist/Metric
        Destination        Gateway         Port          Cost       Type
1       10.0.1.24/32        DIRECT          loopback 1    0/0        D
2       10.30.1.0/24        DIRECT          eth 1/1/7     0/0        D
3       10.40.1.0/24        DIRECT          eth 1/1/2     0/0        D
.
13      10.0.0.6/31       10.0.1.3        eth 1/2/2     20/1       B
14      10.0.0.16/30      10.0.1.3        eth 1/2/2     20/1       B
15      10.0.0.40/29      DIRECT          drop          200/0      B
.          ..               .               .             .          .
42      10.0.0.192/27     DIRECT          drop          200/0      B
43      10.0.1.128/26     10.0.1.3        eth 1/1/7    20/1       B
.          ..               .               .             .          .
69      10.0.7.0/24       10.0.1.3        eth 1/1/10    20/1       B
70      10.0.14.0/23      DIRECT          drop          200/0      B
.          ..               .               .             .          .
.          ..               .               .             .          .
131     10.144.0.0/12     10.0.1.3        eth 1/1/4     20/1       B
132     10.199.1.1/32     DIRECT          drop          1/1        S
```

Modifying redistribution parameters

By default, the Layer 3 Switch does not redistribute route information between BGP4 and the IP IGPs (RIP and OSPF). You can configure the switch to redistribute OSPF routes, RIP routes, directly connected routes, or static routes into BGP4 by using the following methods.

To enable redistribution of all OSPF routes and directly attached routes into BGP4, enter the following commands.

```
Brocade(config)#router bgp
Brocade(config-bgp-router)#redistribute ospf
Brocade(config-bgp-router)#redistribute connected
Brocade(config-bgp-router)#write memory
```

Syntax: [no] **redistribute connected** | **ospf** | **rip** | **static**

The **connected** parameter indicates that you are redistributing routes to directly attached devices into BGP.

The **ospf** parameter indicates that you are redistributing OSPF routes into BGP4.

NOTE
Entering **redistribute ospf** simply redistributes internal OSPF routes. If you want to redistribute external OSPF routes also, you must use the **redistribute ospf match external...** command. Refer to "Redistributing OSPF external routes" on page 331.

The **rip** parameter indicates that you are redistributing RIP routes into BGP4.

The **static** parameter indicates that you are redistributing static routes into BGP.

Refer to the following sections for details on redistributing specific routes using the CLI:

- "Redistributing connected routes" on page 330
- "Redistributing RIP routes" on page 331
- "Redistributing OSPF external routes" on page 331
- "Redistributing static routes" on page 332

Redistributing connected routes

To configure BGP4 to redistribute directly connected routes, enter the following command.

```
Brocade(config-bgp-router)#redistribute connected
```

Syntax: **redistribute connected** [**metric** *num*] [**route-map** *map-name*]

The **connected** parameter indicates that you are redistributing routes to directly attached devices into BGP4.

The **metric** *num* parameter changes the metric. You can specify a value from 0 through 4294967295. The default is 0.

The **route-map** *map-name* parameter specifies a route map to be consulted before adding the RIP route to the BGP4 route table.

NOTE
The route map you specify must already be configured on the switch. Refer to "Defining route maps" on page 342 for information about defining route maps.

Redistributing RIP routes

To configure BGP4 to redistribute RIP routes and add a metric of 10 to the redistributed routes, enter the following command.

```
Brocade(config-bgp-router)#redistribute rip metric 10
```

Syntax: **redistribute rip** [**metric** *num*] [**route-map** *map-name*]

The **rip** parameter indicates that you are redistributing RIP routes into BGP4.

The **metric** *num* parameter changes the metric. Specify a value from 0 through 4294967295. The default is 0.

The **route-map** *map-name* parameter specifies a route map to be consulted before adding the RIP route to the BGP4 route table.

NOTE
The route map you specify must already be configured on the switch. Refer to "Defining route maps" on page 342 for information about defining route maps.

Redistributing OSPF external routes

To configure the Layer 3 switch to redistribute OSPF external type 1 routes, enter the following command.

```
Brocade(config-bgp-router)#redistribute ospf match external1
```

Syntax: **redistribute ospf** [**match internal** | **external1** | **external2**] [**metric** *num*] [**route-map** *map-name*]

The **ospf** parameter indicates that you are redistributing OSPF routes into BGP4.

The **match internal** | **external1** | **external2** parameter applies only to OSPF. This parameter specifies the types of OSPF routes to be redistributed into BGP4. The default is internal.

NOTE
If you do not enter a value for the **match** parameter, (for example, you enter **redistribute ospf** only) then only internal OSPF routes will be redistributed.

The **metric** *num* parameter changes the metric. Specify a value from 0 through 4294967295. The default is 0.

The **route-map** *map-name* parameter specifies a route map to be consulted before adding the OSPF route to the BGP4 route table.

NOTE
The route map you specify must already be configured on the switch. Refer to "Defining route maps" on page 342 for information about defining route maps.

NOTE
If you use both the **redistribute ospf route-map** *map-name* command and the **redistribute ospf match internal** | **external1** | **external2** command, the software uses only the route map for filtering.

Redistributing static routes

To configure the Layer 3 switch to redistribute static routes, enter the following command.

```
Brocade(config-bgp-router)#redistribute static
```

Syntax: **redistribute static** [**metric** *num*] [**route-map** *map-name*]

The **static** parameter indicates that you are redistributing static routes into BGP4.

The **metric** *num* parameter changes the metric. Specify a value from 0 through 4294967295. The default is 0.

The **route-map** *map-name* parameter specifies a route map to be consulted before adding the static route to the BGP4 route table.

NOTE
The route map you specify must already be configured on the switch. Refer to "Defining route maps" on page 342 for information about defining route maps.

Disabling or re-enabling re-advertisement of all learned BGP4 routes to all BGP4 neighbors

By default, the Layer 3 switch re-advertises all learned best BGP4 routes to BGP4 neighbors, unless the routes are discarded or blocked by route maps or other filters.

If you want to prevent the Layer 3 switch from re-advertising a learned best BGP4 route unless that route also is installed in the IP route table, use the following CLI method.

To disable re-advertisement of BGP4 routes to BGP4 neighbors except for routes that the software also installs in the route table, enter the following command.

```
Brocade(config-bgp-router)#no readvertise
```

Syntax: [**no**] **readvertise**

To re-enable re-advertisement, enter the following command.

```
Brocade(config-bgp-router)#readvertise
```

Redistributing IBGP routes into RIP and OSPF

By default, the Layer 3 switch does not redistribute IBGP routes from BGP4 into RIP or OSPF. This behavior helps eliminate routing loops. However, if your network can benefit from redistributing the IBGP routes from BGP4 into OSPF or RIP, you can enable the Layer 3 switch to redistribute the routes. To do so, use the following CLI method.

To enable the Layer 3 switch to redistribute BGP4 routes into OSPF and RIP, enter the following command.

```
Brocade(config-bgp-router)#bgp-redistribute-internal
```

Syntax: [**no**] **bgp-redistribute-internal**

To disable redistribution of IBGP routes into RIP and OSPF, enter the following command.

```
Brocade(config-bgp-router)#no bgp-redistribute-internal
```

Filtering

This section describes the following:

Specific IP address filtering

You can configure the router to explicitly permit or deny specific IP addresses received in updates from BGP4 neighbors by defining IP address filters. The router permits all IP addresses by default. You can define up to 100 IP address filters for BGP4.

- If you want ***permit*** to remain the default behavior, define individual filters to deny specific IP addresses.
- If you want to change the default behavior to ***deny***, define individual filters to permit specific IP addresses.

NOTE
Once you define a filter, the default action for addresses that do not match a filter is "deny". To change the default action to "permit", configure the last filter as "permit any any".

Address filters can be referred to by a BGP neighbor's distribute list number as well as by match statements in a route map.

NOTE
If the filter is referred to by a route map match statement, the filter is applied in the order in which the filter is listed in the match statement.

NOTE
You also can filter on IP addresses by using IP ACLs.

To define an IP address filter to deny routes to 10.157.0.0, enter the following command.

```
Brocade(config-bgp-router)#address-filter 1 deny 10.157.0.0 255.255.0.0
```

Syntax: **address-filter** *num* **permit** | **deny** *ip-addr wildcard mask wildcard*

The *num* parameter is the filter number.

The **permit** | **deny** parameter indicates the action the Layer 3 switch takes if the filter match is true.

- If you specify **permit**, the Layer 3 switch permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the Layer 3 switch denies the route from entering the BGP4 table if the filter match is true.

NOTE
Once you define a filter, the default action for addresses that do not match a filter is “deny”. To change the default action to “permit”, configure the last filter as “permit any any”.

The *ip-addr* parameter specifies the IP address. If you want the filter to match on all addresses, enter **any**.

The *wildcard* parameter specifies the portion of the IP address to match against. The *wildcard* is in dotted-decimal notation (IP address format). It is a four-part value, where each part is 8 bits (one byte) separated by dots, and each bit is a one or a zero. Each part is a number ranging from 0 to 255, for example 0.0.0.255. Zeros in the mask mean the packet source address must match the *source-ip*. Ones mean any value matches. For example, the *ip-addr* and *wildcard* values 10.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 10.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “10.157.22.26 0.0.0.255” as “10.157.22.26/24”. The CLI automatically converts the CIDR number into the appropriate mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros. For example, if you specify 10.157.22.26/24 or 10.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 10.157.22.0/24 (if you have enabled display of subnet lengths) or 10.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in “/*mask-bits*” format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the filter regardless of whether the software is configured to display the masks in CIDR format.

The *mask* parameter specifies the network mask. If you want the filter to match on all destination addresses, enter **any**. The wildcard works the same as described above.

AS-path filtering

You can filter updates received from BGP4 neighbors based on the contents of the AS-path list accompanying the updates. For example, if you want to deny routes that have the AS 4.3.2.1 in the AS-path from entering the BGP4 route table, you can define a filter to deny such routes.

The Layer 3 switch provides the following methods for filtering on AS-path information:

- AS-path filters
- AS-path ACLs

NOTE
The Layer 3 switch cannot actively support AS-path filters and AS-path ACLs at the same time. Use one method or the other but do not mix methods.

NOTE
Once you define a filter or ACL, the default action for updates that do not match a filter is “deny”. To change the default action to “permit”, configure the last filter or ACL as “permit any any”.

AS-path filters or AS-path ACLs can be referred to by a BGP neighbor's filter list number as well as by match statements in a route map.

Defining an AS-path filter

To define AS-path filter 4 to permit AS 2500, enter the following command.

```
Brocade(config-bgp-router)#as-path-filter 4 permit 2500
```

Syntax: **as-path-filter** *num* **permit** | **deny** *as-path*

The *num* parameter identifies the filter position in the AS-path filter list and can be from 1 through 100. Thus, the AS-path filter list can contain up to 100 filters. The Brocade Layer 3 switch applies the filters in numerical order, beginning with the lowest-numbered filter. When a filter match is true, the Layer 3 switch stops and does not continue applying filters from the list.

NOTE
If the filter is referred to by a route map match statement, the filter is applied in the order in which the filter is listed in the match statement.

The **permit** | **deny** parameter indicates the action the router takes if the filter match is true.

- If you specify **permit**, the router permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the router denies the route from entering the BGP4 table if the filter match is true.

The *as-path* parameter indicates the AS-path information. You can enter an exact AS-path string if you want to filter for a specific value. You also can use regular expressions in the filter string.

Defining an AS-path ACL

To configure an AS-path list that uses ACL 1, enter a command such as the following.

```
Brocade(config)#ip as-path access-list 1 permit 100
Brocade(config)#router bgp
Brocade(config-bgp-router)#neighbor 10.10.10.1 filter-list 1 in
```

The **ip as-path** command configures an AS-path ACL that permits routes containing AS number 100 in their AS paths. The **neighbor** command then applies the AS-path ACL to advertisements and updates received from neighbor 10.10.10.1. In this example, the only routes the Layer 3 switch permits from neighbor 10.10.10.1 are those whose AS-paths contain AS-path number 100.

Syntax: **ip as-path access-list** *string* [**seq** *seq-value*] **deny** | **permit** *regular-expression*

The *string* parameter specifies the ACL name. (If you enter a number, the CLI interprets the number as a text string.)

The **seq** *seq-value* parameter is optional and specifies the AS-path list sequence number. You can configure up to 199 entries in an AS-path list. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with number 5. The software interprets the entries in an AS-path list in numerical order, beginning with the lowest sequence number.

The **deny** | **permit** parameter specifies the action the software takes if a route AS-path list matches a match statement in this ACL. To configure the AS-path match statements in a route map, use the **match as-path** command. Refer to "Matching based on AS-path ACL" on page 345.

The *regular-expression* parameter specifies the AS path information you want to permit or deny to routes that match any of the match statements within the ACL. You can enter a specific AS number or use a regular expression. For the regular expression syntax, refer to "Using regular expressions to filter" on page 336.

The **neighbor** command uses the **filter-list** parameter to apply the AS-path ACL to the neighbor. Refer to "Adding BGP4 neighbors" on page 292.

Using regular expressions to filter

You use a regular expression for the *as-path* parameter to specify a single character or multiple characters as a filter pattern. If the AS-path matches the pattern specified in the regular expression, the filter evaluation is true; otherwise, the evaluation is false.

In addition, you can include special characters that influence the way the software matches the AS-path against the filter value.

To filter on a specific single-character value, enter the character for the *as-path* parameter. For example, to filter on AS-paths that contain the letter "z", enter the following command.

```
Brocade(config-bgp-router)#as-path-filter 1 permit z
```

To filter on a string of multiple characters, enter the characters in brackets. For example, to filter on AS-paths that contain "x", "y", or "z", enter the following command.

```
Brocade(config-bgp-router)#as-path-filter 1 permit [xyz]
```

BGP4 special characters

When you enter as single-character expression or a list of characters, you also can use the following special characters. Table 62 on page 336 lists the special characters. The description for each special character includes an example. Notice that you place some special characters in front of the characters they control but you place other special characters after the characters they control. In each case, the examples show where to place the special character.

TABLE 62 BGP4 special characters for regular expressions

Character	Operation
.	The period matches on any single character, including a blank space. For example, the following regular expression matches for "aa", "ab", "ac", and so on, but not just "a". a.
*	The asterisk matches on zero or more sequences of a pattern. For example, the following regular expression matches on an AS-path that contains the string "1111" followed by any value: 1111*
+	The plus sign matches on one or more sequences of a pattern. For example, the following regular expression matches on an AS-path that contains a sequence of "g"s, such as "deg", "degg", "deggg", and so on: deg+
?	The question mark matches on zero occurrences or one occurrence of a pattern. For example, the following regular expression matches on an AS-path that contains "dg" or "deg": de?g
^	A caret (when not used within brackets) matches on the beginning of an input string. For example, the following regular expression matches on an AS-path that begins with "3": ^3
$	A dollar sign matches on the end of an input string. For example, the following regular expression matches on an AS-path that ends with "deg": deg$

TABLE 62 BGP4 special characters for regular expressions (Continued)

Character	Operation
_	An underscore matches on one or more of the following: • , (comma) • { (left curly brace) • } (right curly brace) • ((left parenthesis) •) (right parenthesis) • The beginning of the input string • The end of the input string • A blank space For example, the following regular expression matches on "100" but not on "1002", "2100", and so on. _100_
[ ]	Square brackets enclose a range of single-character patterns. For example, the following regular expression matches on an AS-path that contains "1", "2", "3", "4", or "5": [1-5] You can use the following expression symbols within the brackets. These symbols are allowed only inside the brackets: • ^ – The caret matches on any characters except the ones in the brackets. For example, the following regular expression matches on an AS-path that does not contain "1", "2", "3", "4", or "5": [^1-5] • - The hyphen separates the beginning and ending of a range of characters. A match occurs if any of the characters within the range is present. See the example above.
|	A vertical bar (sometimes called a pipe or a "logical or") separates two alternative values or sets of values. The AS-path can match one or the other value. For example, the following regular expression matches on an AS-path that contains either "abc" or "defg": (abc)|(defg) **NOTE:** The parentheses group multiple characters to be treated as one value. See the following row for more information about parentheses.
( )	Parentheses allow you to create complex expressions. For example, the following complex expression matches on "abc", "abcabc", or "abcabcabcdefg", but not on "abcdefgdefg": ((abc)+)|((defg)?)

If you want to filter for a special character instead of using the special character as described in Table 62 on page 336, enter "\" (backslash) in front of the character. For example, to filter on AS-path strings containing an asterisk, enter the asterisk portion of the regular expression as "*".

```
Brocade(config-bgp-router)#as-path-filter 2 deny \*
```

To use the backslash as a string character, enter two slashes. For example, to filter on AS-path strings containing a backslash, enter the backslash portion of the regular expression as "\\".

```
Brocade(config-bgp-router)#as-path-filter 2 deny \\
```

BGP4 filtering communities

You can filter routes received from BGP4 neighbors based on community names. Use either of the following methods to do so.

A community is an optional attribute that identifies the route as a member of a user-defined class of routes. Community names are arbitrary values made of two five-digit integers joined by a colon. You determine what the name means when you create the community name as one of a route attributes. Each string in the community name can be a number from 0 through 65535.

This format allows you to easily classify community names. For example, a common convention used in community naming is to configure the first string as the local AS and the second string as the unique community within that AS. Using this convention, communities 1:10, 1:20, and 1:30 can be easily identified as member communities of AS 1.

The Layer 3 switch provides the following methods for filtering on community information:

- Community filters
- Community list ACLs

NOTE
The Layer 3 switch cannot actively support community filters and community list ACLs at the same time. Use one method or the other but do not mix methods.

NOTE
Once you define a filter or ACL, the default action for communities that do not match a filter or ACL is “deny”. To change the default action to “permit”, configure the last filter or ACL entry as “permit any any”.

Community filters or ACLs can be referred to by match statements in a route map.

Defining a community filter

To define filter 3 to permit routes that have the NO_ADVERTISE community, enter the following command.

```
Brocade(config-bgp-router)#community-filter 3 permit no-advertise
```

Syntax: **community-filter** *num* **permit** | **deny** *num:num* | **internet** | **local-as** | **no-advertise** | **no-export**

The *num* parameter identifies the filter position in the community filter list and can be from 1 through 100. Thus, the community filter list can contain up to 100 filters. The router applies the filters in numerical order, beginning with the lowest-numbered filter. When a filter match is true, the router stops and does not continue applying filters from the list.

NOTE
If the filter is referred to by a route map match statement, the filter is applied in the order in which the filter is listed in the match statement.

The **permit** | **deny** parameter indicates the action the router takes if the filter match is true.

- If you specify **permit**, the router permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the router denies the route from entering the BGP4 table if the filter match is true.

The *num:num* parameter indicates a specific community number to filter. Use this parameter to filter for a private (administrator-defined) community. You can enter up to 20 community numbers with the same command.

If you want to filter for the well-known communities “LOCAL_AS”, “NO_EXPORT” or “NO_ADVERTISE”, use the corresponding keyword (described below).

The **internet** keyword checks for routes that do not have the community attribute. Routes without a specific community are considered by default to be members of the largest community, the Internet.

The **local-as** keyword checks for routes with the well-known community “LOCAL_AS”. This community applies only to confederations. The Layer 3 switch advertises the route only within the sub-AS. For information about confederations, refer to “Configuration notes for BGP4 autonomous systems” on page 320.

The **no-advertise** keyword filters for routes with the well-known community “NO_ADVERTISE”. A route in this community should not be advertised to any BGP4 neighbors.

The **no-export** keyword filters for routes with the well-known community “NO_EXPORT”. A route in this community should not be advertised to any BGP4 neighbors outside the local AS. If the router is a member of a confederation, the Layer 3 switch advertises the route only within the confederation. For information about confederations, refer to “Configuration notes for BGP4 autonomous systems” on page 320.

Defining a community ACL

To configure community ACL 1, enter a command such as the following.

```
Brocade(config)#ip community-list 1 permit 123:2
```

This command configures a community ACL that permits routes that contain community 123:2.

NOTE
Refer to “Matching based on community ACL” on page 345 for information about how to use a community list as a match condition in a route map.

Syntax: **ip community-list standard** *string* [**seq** *seq-value*] **deny** | **permit** *community-num*

Syntax: **ip community-list extended** *string* [**seq** *seq-value*] **deny** | **permit** *community-num* | *regular-expression*

The *string* parameter specifies the ACL name. (If you enter a number, the CLI interprets the number as a text string.)

The **standard** or **extended** parameter specifies whether you are configuring a standard community ACL or an extended one. A standard community ACL does not support regular expressions whereas an extended one does. This is the only difference between standard and extended IP community lists.

The **seq** *seq-value* parameter is optional and specifies the community list sequence number. You can configure up to 199 entries in a community list. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with number 5. The software interprets the entries in a community list in numerical order, beginning with the lowest sequence number.

The **deny** | **permit** parameter specifies the action the software takes if a route community list matches a match statement in this ACL. To configure the community-list match statements in a route map, use the **match community** command. Refer to "Matching based on community ACL" on page 345.

The *community-num* parameter specifies the community type or community number. This parameter can have the following values:

- *num:num* – A specific community number
- **internet** – The Internet community
- **no-export** – The community of sub-autonomous systems within a confederation. Routes with this community can be exported to other sub-autonomous systems within the same confederation but cannot be exported outside the confederation to other autonomous systems or otherwise sent to EBGP neighbors.
- **local-as** – The local sub-AS within the confederation. Routes with this community can be advertised only within the local subAS.
- **no-advertise** – Routes with this community cannot be advertised to any other BGP4 routers at all.

The *regular-expression* parameter specifies a regular expression for matching on community names. For information about regular expression syntax, refer to "Using regular expressions to filter" on page 336. You can specify a regular expression only in an extended community ACL.

Defining IP prefix lists

An IP prefix list specifies a list of networks. When you apply an IP prefix list to a neighbor, the Layer 3 switch sends or receives only a route whose destination is in the IP prefix list. You can configure up to 100 prefix lists. The software interprets the prefix lists in order, beginning with the lowest sequence number.

To configure an IP prefix list and apply it to a neighbor, enter commands such as the following.

```
Brocade(config)#ip prefix-list Routesfor20 permit 10.20.0.0/24
Brocade(config)#router bgp
Brocade(config-bgp-router)#neighbor 10.10.10.1 prefix-list Routesfor20 out
```

These commands configure an IP prefix list named Routesfor20, which permits routes to network 10.20.0.0/24. The **neighbor** command configures the Layer 3 switch to use IP prefix list Routesfor20 to determine which routes to send to neighbor 10.10.10.1. The Layer 3 switch sends routes that go to 10.20.x.x to neighbor 10.10.10.1 because the IP prefix list explicitly permits these routes to be sent to the neighbor.

Syntax: **ip prefix-list** *name* [**seq** *seq-value*] [**description** *string*] **deny** | **permit** *network-addr/mask-bits* [**ge** *ge-value*] [**le** *le-value*]

The *name* parameter specifies the prefix list name. You use this name when applying the prefix list to a neighbor.

The **description** *string* parameter is a text string describing the prefix list.

The **seq** *seq-value* parameter is optional and specifies the IP prefix list sequence number. You can configure up to 100 prefix list entries. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with prefix list entry 5. The software interprets the prefix list entries in numerical order, beginning with the lowest sequence number.

The **deny** | **permit** parameter specifies the action the software takes if a neighbor route is in this prefix list.

The prefix-list matches only on this network unless you use the **ge** *ge-value* or **le** *le-value* parameters. (See below.)

The *network-addr/mask-bits* parameter specifies the network number and the number of bits in the network mask.

You can specify a range of prefix length for prefixes that are more specific than *network-addr/mask-bits*.

- If you specify only **ge** *ge-value*, then the mask-length range is from *ge-value* to 32.
- If you specify only **le** *le-value*, then the mask-length range is from length to *le-value*.

The *ge-value* or *le-value* you specify must meet the following condition.

length < ge-value <= le-value <= 32

If you do not specify **ge** *ge-value* or **le** *le-value*, the prefix list matches only on the exact network prefix you specify with the *network-addr/mask-bits* parameter.

For the syntax of the **neighbor** command shown in the example above, refer to "Adding BGP4 neighbors" on page 292.

Defining neighbor distribute lists

A neighbor distribute list is a list of BGP4 address filters or ACLs that filter the traffic to or from a neighbor. To configure a neighbor distribute list, use either of the following methods.

To configure a distribute list that uses ACL 1, enter a command such as the following.

```
Brocade(config-bgp-router)#neighbor 10.10.10.1 distribute-list 1 in
```

This command configures the Layer 3 switch to use ACL 1 to select the routes that the Layer 3 switch will accept from neighbor 10.10.10.1.

Syntax: **neighbor** *ip-addr* **distribute-list** *name-or-num* **in** | **out**

The *ip-addr* parameter specifies the neighbor.

The *name-or-num* parameter specifies the name or number of a standard, extended, or named ACL.

The **in** | **out** parameter specifies whether the distribute list applies to inbound or outbound routes:

- **in** – controls the routes the Layer 3 switch will accept from the neighbor.
- **out** – controls the routes sent to the neighbor.

NOTE
The command syntax shown above is new. However, the **neighbor** *ip-addr* **distribute-list in** | **out** *num* command (where the direction is specified before the filter number) is the same as in earlier software releases. Use the new syntax when you are using an IP ACL with the distribute list. Use the old syntax when you are using a BGP4 address filter with the distribute list.

Defining route maps

A ***route map*** is a named set of match conditions and parameter settings that the router can use to modify route attributes and to control redistribution of the routes into other protocols. A route map consists of a sequence of up to 50 ***instances***. If you think of a route map as a table, an instance is a row in that table. The router evaluates a route according to a route map instances in ascending numerical order. The route is first compared against instance 1, then against instance 2, and so on. As soon as a match is found, the router stops evaluating the route against the route map instances.

Route maps can contain ***match*** statements and ***set*** statements. Each route map contains a "permit" or "deny" action for routes that match the match statements:

- If the route map contains a permit action, a route that matches a match statement is permitted; otherwise, the route is denied.
- If the route map contains a deny action, a route that matches a match statement is denied.
- If a route does not match any match statements in the route map, the route is denied. This is the default action. To change the default action, configure the last match statement in the last instance of the route map to "permit any any".
- If there is no match statement, the software considers the route to be a match.
- For route maps that contain address filters, AS-path filters, or community filters, if the action specified by a filter conflicts with the action specified by the route map, the route map action takes precedence over the individual filter action.

If the route map contains set statements, routes that are permitted by the route map match statements are modified according to the set statements.

Match statements compare the route against one or more of the following:

- The route BGP4 MED (metric)
- A sequence of AS-path filters
- A sequence of community filters
- A sequence of address filters
- The IP address of the next hop router
- The route tag
- For OSPF routes only, the route type (internal, external type-1, or external type-2)
- An AS-path ACL
- A community ACL
- An IP prefix list
- An IP ACL

For routes that match all of the match statements, the route map set statements can perform one or more of the following modifications to the route attributes:

- Prepend AS numbers to the front of the route AS-path. By adding AS numbers to the AS-path, you can cause the route to be less preferred when compared to other routes on the basis of the length of the AS-path.
- Add a user-defined tag to the route or add an automatically calculated tag to the route.
- Set the community value.
- Set the local preference.

- Set the MED (metric).
- Set the IP address of the next hop router.
- Set the origin to IGP or INCOMPLETE.
- Set the weight.

For example, when you configure parameters for redistributing routes into RIP, one of the optional parameters is a route map. If you specify a route map as one of the redistribution parameters, the router will match the route against the match statements in the route map. If a match is found and if the route map contains set statements, the router will set attributes in the route according to the set statements.

To create a route map, you define instances of the map. Each instance is identified by a sequence number. A route map can contain up to 50 instances.

To define a route map, use the procedures in the following sections.

Entering the route map into the software

To add instance 1 of a route map named "GET_ONE" with a permit action, enter the following command.

```
Brocade(config)#route-map GET_ONE permit 1
Brocade(config-routemap GET_ONE)#
```

Syntax: [**no**] **route-map** *map-name* **permit** | **deny** *num*

As shown in this example, the command prompt changes to the Route Map level.   You can enter the match and set statements at this level. Refer to "Specifying the match conditions" on page 344 and "Setting parameters in the routes" on page 347.

The *map-name* is a string of characters that names the map. Map names can be up to 32 characters in length.

The **permit** | **deny** parameter specifies the action the router will take if a route matches a match statement.

- If you specify **deny**, the Layer 3 switch does not advertise or learn the route.
- If you specify **permit**, the Layer 3 switch applies the match and set statements associated with this route map instance.

The *num* parameter specifies the instance of the route map you are defining. Each route map can have up to 50 instances.

To delete a route map, enter a command such as the following. When you delete a route map, all the permit and deny entries in the route map are deleted.

```
Brocade(config)#no route-map Map1
```

This command deletes a route map named "Map1". All entries in the route map are deleted.

To delete a specific instance of a route map without deleting the rest of the route map, enter a command such as the following.

```
Brocade(config)#no route-map Map1 permit 10
```

This command deletes the specified instance from the route map but leaves the other instances of the route map intact.

Specifying the match conditions

Use the following command to define the match conditions for instance 1 of the route map GET_ONE. This instance compares the route updates against BGP4 address filter 11.

```
Brocade(config-routemap GET_ONE)#match address-filters 11
```

Syntax: **match** [**as-path** *num*] | [**address-filters** | **as-path-filters** | **community-filters** *num,num,..*] | [**community** *num*] | [**community** *ACL* **exact-match**] | [**ip address** *ACL* | **prefix-list** *string*] | [**ip route-source** *ACL* | **prefix** *name*] [**metric** *num*] | [**next-hop** *address-filter-list*] | [**nlri multicast** | **unicast** | **multicast unicast**] | [**route-type internal** | **external-type1** | **external-type2**] | [**tag** *tag-value*]

The **as-path** *num* parameter specifies an AS-path ACL. You can specify up to five AS-path ACLs. To configure an AS-path ACL, use the **ip as-path access-list** command. Refer to "Defining an AS-path ACL" on page 335.

The **address-filters** | **as-path-filters** | **community-filters** *num,num,...* parameter specifies a filter or list of filters to be matched for each route. The router treats the first match as the best match. If a route does not match any filter in the list, then the router considers the match condition to have failed. To configure these types of filters, use commands at the BGP configuration level:

- To configure an address filter, refer to "Specific IP address filtering" on page 333.
- To configure an AS-path filter or AS-path ACL, refer to "AS-path filtering" on page 334.
- To configure a community filter or community ACL, refer to "BGP4 filtering communities" on page 338.

You can enter up to six community names on the same command line.

NOTE
The filters must already be configured.

The **community** *num* parameter specifies a community ACL.

NOTE
The ACL must already be configured.

The **community** *ACL* **exact-match** parameter matches a route if (and only if) the route's community attributes field contains the same community numbers specified in the match statement.

The **ip address** | **next-hop** *ACL-num* | prefix-list *string* parameter specifies an ACL or IP prefix list. Use this parameter to match based on the destination network or next-hop gateway. To configure an IP ACL for use with this command, use the **ip access-list** command. Refer to the section "ACL overview" in the Brocade ICX 6650 *Security Configuration Guide*. To configure an IP prefix list, use the **ip prefix-list** command. Refer to "Defining IP prefix lists" on page 340.

The **ip route-source** *ACL* | **prefix** *name* parameter matches based on the source of a route (the IP address of the neighbor from which the Brocade device learned the route).

The **metric** *num* parameter compares the route MED (metric) to the specified value.

The **next-hop** *address-filter-list* parameter compares the IP address of the route next hop to the specified IP address filters. The filters must already be configured.

The **nlri multicast** | **unicast** | **multicast unicast** parameter specifies whether you want the route map to match on multicast routes, unicast routes, or both route types.

NOTE
By default, route maps apply to both unicast and multicast traffic.

The **route-type internal** | **external-type1** | **external-type2** parameter applies only to OSPF routes. This parameter compares the route type to the specified value.

The **tag** *tag-value* parameter compares the route tag to the specified value.

Match examples using ACLs

The following sections show some detailed examples of how to configure route maps that include match statements that match on ACLs.

Matching based on AS-path ACL

To construct a route map that matches based on AS-path ACL 1, enter the following commands.

```
Brocade(config)#route-map PathMap permit 1
Brocade(config-routemap PathMap)#match as-path 1
```

Syntax: **match as-path** *num*

The *num* parameter specifies an AS-path ACL and can be a number from 1 through 199. You can specify up to five AS-path ACLs. To configure an AS-path ACL, use the **ip as-path access-list** command. Refer to "Defining an AS-path ACL" on page 335.

Matching based on community ACL

To construct a route map that matches based on community ACL 1, enter the following commands.

```
Brocade(config)#ip community-list 1 permit 123:2
Brocade(config)#route-map CommMap permit 1
Brocade(config-routemap CommMap)#match community 1
```

Syntax: **match community** *string*

The *string* parameter specifies a community list ACL. To configure a community list ACL, use the **ip community-list** command. Refer to "Defining a community ACL" on page 339.

Matching based on destination network

To construct match statements for a route map that match based on destination network, use the following method. You can use the results of an IP ACL or an IP prefix list as the match condition.

```
Brocade(config)#route-map NetMap permit 1
Brocade(config-routemap NetMap)#match ip address 1
```

Syntax: **match ip address** *name-or-num*

Syntax: **match ip address prefix-list** *name*

The *name-or-num* parameter with the first command specifies an IP ACL and can be a number from 1 through 199 or the ACL name if it is a named ACL. To configure an IP ACL, use the **ip access-list** or **access-list** command. Refer to the chapter "Rule-Based IP ACLs" in the *Brocade ICX 6650 Security Configuration Guide*.

The *name* parameter with the second command specifies an IP prefix list name. To configure an IP prefix list, refer to "Defining IP prefix lists" on page 340.

Matching based on next-hop router

To construct match statements for a route map that match based on the IP address of the next-hop router, use either of the following methods. You can use the results of an IP ACL or an IP prefix list as the match condition.

To construct a route map that matches based on the next-hop router, enter commands such as the following.

```
Brocade(config)#route-map HopMap permit 1
Brocade(config-routemap HopMap)#match ip next-hop 2
```

Syntax: **match ip next-hop** *num*

Syntax: **match ip next-hop prefix-list** *name*

The *num* parameter with the first command specifies an IP ACL and can be a number from 1 through 199 or the ACL name if it is a named ACL. To configure an IP ACL, use the **ip access-list** or **access-list** command. Refer to the chapter "Rule-Based IP ACLs" in the *Brocade ICX 6650 Security Configuration Guide*.

The *name* parameter with the second command specifies an IP prefix list name. To configure an IP prefix list, refer to "Defining IP prefix lists" on page 340.

Matching based on the route source

To match a BGP4 route based on its source, use the **match ip route-source** statement. Here is an example.

```
Brocade(config)#access-list 10 permit 192.168.6.0 0.0.0.255
Brocade(config)#route-map bgp1 permit 1
Brocade(config-routemap bgp1)#match ip route-source 10
```

The first command configures an IP ACL that matches on routes received from 192.168.6.0/24. The remaining commands configure a route map that matches on all BGP4 routes advertised by the BGP4 neighbors whose addresses match addresses in the IP prefix list. You can add a set statement to change a route attribute in the routes that match. You also can use the route map as input for other commands, such as the **neighbor** and **network** commands and some **show** commands.

Syntax: **match ip route-source** *ACL* | **prefix** *name*

The *ACL* | prefix *name* parameter specifies the name or ID of an IP ACL, or an IP prefix list.

Matching on routes containing a specific set of communities

Brocade software enables you to match routes based on the presence of a community name or number in a route, and to match when a route contains exactly the set of communities you specify. To match based on a set of communities, configure a community ACL that lists the communities, then compare routes against the ACL, as shown in the following example.

```
Brocade(config)#ip community-list standard std_1 permit 12:34 no-export
Brocade(config)#route-map bgp2 permit 1
Brocade(config-routemap bgp2)#match community std_1 exact-match
```

The first command configures a community ACL that contains community number 12:34 and community name no-export. The remaining commands configure a route map that matches the community attributes field in BGP4 routes against the set of communities in the ACL. A route matches the route map only if the route contains all the communities in the ACL and no other communities.

Syntax: **match community** *ACL* **exact-match**

The *ACL* parameter specifies the name of a community list ACL. You can specify up to five ACLs. Separate the ACL names or IDs with spaces.

Here is another example.

```
Brocade(config)#ip community-list standard std_2 permit 23:45 56:78
Brocade(config)#route-map bgp3 permit 1
Brocade(config-routemap bgp3)#match community std_1 std_2 exact-match
```

These commands configure an additional community ACL, std_2, that contains community numbers 23:45 and 57:68. Route map bgp3 compares each BGP4 route against the sets of communities in ACLs std_1 and std_2. A BGP4 route that contains ***either but not both*** sets of communities matches the route map. For example, a route containing communities 23:45 and 57:68 matches. However, a route containing communities 23:45, 57:68 and 12:34, or communities 23:45, 57:68, 12:34, and no-export does not match. To match, the route communities must be the same as those in exactly one of the community ACLs used by the match community statement.

Setting parameters in the routes

Use the following command to define a set statement that prepends an AS number to the AS path on each route that matches the corresponding match statement.

```
Brocade(config-routemap GET_ONE)#set as-path prepend 65535
```

Syntax: **set** [**as-path** [**prepend** *as-num,as-num,...*]] | [**automatic-tag**] | [**comm-list** *ACL* **delete**] | [**community** *num:num* | *num* | **internet** | **local-as** | **no-advertise** | **no-export**] | [**dampening** [*half-life reuse suppress max-suppress-time*]] [[**default**] **interface null0** | [**ip** [**default**] **next hop** *ip-addr*] [**ip next-hop peer-address**] | [**local-preference** *num*] | [**metric** [**+** | **-**]*num* | **none**] | [**metric-type type-1** | **type-2**] | [**metric-type internal**] | [**next-hop** *ip-addr*] | [**nlri multicast** | **unicast** | **multicast unicast**] | [**origin igp** | **incomplete**] | [**tag** *tag-value*] | [**weight** *num*]

The **as-path prepend** *num,num,...* parameter adds the specified AS numbers to the front of the AS-path list for the route.

The **automatic-tag** parameter calculates and sets an automatic tag value for the route.

NOTE
This parameter applies only to routes redistributed into OSPF.

The **comm-list** parameter deletes a community from a BGP4 route community attributes field.

The **community** parameter sets the community attribute for the route to the number or well-known type you specify.

The **dampening** [*half-life reuse suppress max-suppress-time*] parameter sets route dampening parameters for the route. The *half-life* parameter specifies the number of minutes after which the route penalty becomes half its value. The *reuse* parameter specifies how low a route penalty must become before the route becomes eligible for use again after being suppressed. The *suppress* parameter specifies how high a route penalty can become before the Layer 3 switch suppresses the route. The *max-suppress-time* parameter specifies the maximum number of minutes that a route can be suppressed regardless of how unstable it is. For information and examples, refer to "Route flap dampening configuration" on page 354.

The [**default**] **interface null0** parameter redirects the traffic to the specified interface. You can send the traffic to the null0 interface, which is the same as dropping the traffic. You can specify more than one interface, in which case the Layer 3 switch uses the first available port. If the first port is unavailable, the Layer 3 switch sends the traffic to the next port in the list. If you specify **default**, the route map redirects the traffic to the specified interface only if the Layer 3 switch does not already have explicit routing information for the traffic. This option is used in Policy-Based Routing (PBR).

The **ip** [**default**] **next hop** *ip-addr* parameter sets the next-hop IP address for traffic that matches a match statement in the route map. If you specify **default**, the route map sets the next-hop gateway only if the Layer 3 switch does not already have explicit routing information for the traffic. This option is used in Policy-Based Routing (PBR).

The **ip next-hop peer-address** parameter sets the BGP4 next hop for a route to the specified neighbor address.

The **local-preference** *num* parameter sets the local preference for the route. You can set the preference to a value from 0 through 4294967295.

The **metric** [+ | -]*num* | none parameter sets the MED (metric) value for the route. The default MED value is 0. You can set the preference to a value from 0 through 4294967295.

- **set metric** *num* – Sets the route metric to the number you specify.
- **set metric** +*num* – Increases route metric by the number you specify.
- **set metric** -*num* – Decreases route metric by the number you specify.
- **set metric none** – Removes the metric from the route (removes the MED attribute from the BGP4 route).

The **metric-type type-1** | **type-2** parameter changes the metric type of a route redistributed into OSPF.

The **metric-type internal** parameter sets the route's MED to the same value as the IGP metric of the BGP4 next-hop route. The parameter does this when advertising a BGP4 route to an EBGP neighbor.

The **next-hop** *ip-addr* parameter sets the IP address of the route next hop router.

The **nlri multicast** | **unicast** | **multicast unicast** parameter redistributes routes into the multicast Routing Information Base (RIB) instead of the unicast RIB.

NOTE
Setting the NLRI type to multicast applies only when you are using the route map to redistribute directly-connected routes. Otherwise, the set option is ignored.

The **origin igp** | **incomplete** parameter sets the route origin to IGP or INCOMPLETE.

The **tag** *tag-value* parameter sets the route tag. You can specify a tag value from 0 through 4294967295.

NOTE
This parameter applies only to routes redistributed into OSPF.

NOTE
You also can set the tag value using a table map. The table map changes the value only when the Layer 3 switch places the route in the IP route table instead of changing the value in the BGP route table. Refer to "Using a table map to set the tag value" on page 350.

The **weight** *num* parameter sets the weight for the route. You can specify a weight value from 0 through 4294967295.

Setting a BP4 route MED to the same value as the IGP metric of the next-hop route

To set a route's MED to the same value as the IGP metric of the BGP4 next-hop route, when advertising the route to a neighbor, enter commands such as the following.

```
Brocade(config)#access-list 1 permit 192.168.9.0 0.0.0.255
Brocade(config)#route-map bgp4 permit 1
Brocade(config-routemap bgp4)#match ip address 1
Brocade(config-routemap bgp4)#set metric-type internal
```

The first command configures an ACL that matches on routes with destination network 192.168.9.0. The remaining commands configure a route map that matches on the destination network in ACL 1, then sets the metric type for those routes to the same value as the IGP metric of the BGP4 next-hop route.

Syntax: set metric-type internal

Setting the next hop of a BGP4 route

To set the next hop address of a BGP4 route to a neighbor address, enter commands such as the following.

```
Brocade(config)#route-map bgp5 permit 1
Brocade(config-routemap bgp5)#match ip address 1
Brocade(config-routemap bgp5)#set ip next-hop peer-address
```

These commands configure a route map that matches on routes whose destination network is specified in ACL 1, and sets the next hop in the routes to the neighbor address (inbound filtering) or the local IP address of the BGP4 session (outbound filtering).

Syntax: set ip next-hop peer-address

The value that the software substitutes for **peer-address** depends on whether the route map is used for inbound filtering or outbound filtering:

- When you use the **set ip next-hop peer-address** command in an inbound route map filter, **peer-address** substitutes for the neighbor IP address.
- When you use the **set ip next-hop peer-address** command in an outbound route map filter, **peer-address** substitutes for the local IP address of the BGP4 session.

NOTE
You can use this command for a peer group configuration.

Deleting a community from a BGP4 route

To delete a community from a BGP4 route community attributes field, enter commands such as the following.

```
Brocade(config)#ip community-list standard std_3 permit 12:99 12:86
Brocade(config)#route-map bgp6 permit 1
Brocade(config-routemap bgp6)#match ip address 1
Brocade(config-routemap bgp6)#set comm-list std_3 delete
```

The first command configures a community ACL containing community numbers 12:99 and 12:86. The remaining commands configure a route map that matches on routes whose destination network is specified in ACL 1, and deletes communities 12:99 and 12:86 from those routes. The route does not need to contain all the specified communities in order for them to be deleted. For example, if a route contains communities 12:86, 33:44, and 66:77, community 12:86 is deleted.

Syntax: **set comm-list** *ACL* **delete**

The *ACL* parameter specifies the name of a community list ACL.

Using a table map to set the tag value

Route maps that contain set statements change values in routes when the routes are accepted by the route map. For inbound route maps (route maps that filter routes received from neighbors), this means that the routes are changed before they enter the BGP4 route table.

For tag values, if you do not want the value to change until a route enters the IP route table, you can use a table map to change the value. A table map is a route map that you have associated with the IP routing table. The Layer 3 switch applies the set statements for tag values in the table map to routes before adding them to the route table.

To configure a table map, you configure the route map, then identify it as a table map. The table map does not require separate configuration. You create it simply by calling an existing route map a table map. You can have one table map.

NOTE
Use table maps only for setting the tag value. Do not use table maps to set other attributes. To set other route attributes, use route maps or filters.

To create a route map and identify it as a table map, enter commands such as following. These commands create a route map that uses an address filter. For routes that match the address filter, the route map changes the tag value to 100. This route map is then identified as a table map. As a result, the route map is applied only to routes that the Layer 3 switch places in the IP route table. The route map is not applied to all routes. This example assumes that address filter 11 has already been configured.

```
Brocade(config)#route-map TAG_IP permit 1
Brocade(config-routemap TAG_IP)#match address-filters 11
Brocade(config-routemap TAG_IP)#set tag 100
Brocade(config-routemap TAG_IP)#router bgp
Brocade(config-bgp-router)#table-map TAG_IP
```

Configuring cooperative BGP4 route filtering

By default, the Layer 3 switch performs all filtering of incoming routes locally, on the Layer 3 switch itself. You can use cooperative BGP4 route filtering to cause the filtering to be performed by a neighbor before it sends the routes to the Layer 3 switch. Cooperative filtering conserves resources by eliminating unnecessary route updates and filter processing. For example, the Layer 3 switch can send a deny filter to its neighbor, which the neighbor uses to filter out updates before sending them to the Layer 3 switch. The neighbor saves the resources it would otherwise use to generate the route updates, and the Layer 3 switch saves the resources it would use to filter out the routes.

When you enable cooperative filtering, the Layer 3 switch advertises this capability in its Open message to the neighbor when initiating the neighbor session. The Open message also indicates whether the Layer 3 switch is configured to send filters, receive filters or both, and the types of filters it can send or receive. The Layer 3 switch sends the filters as Outbound Route Filters (ORFs) in Route Refresh messages.

To configure cooperative filtering, perform the following tasks on the Layer 3 switch and on its BGP4 neighbor:

- Configure the filter.

  NOTE
  The current release supports cooperative filtering only for filters configured using IP prefix lists.

- Apply the filter as in *inbound* filter to the neighbor.
- Enable the cooperative route filtering feature on the Layer 3 switch. You can enable the Layer 3 switch to send ORFs to the neighbor, to receive ORFs from the neighbor, or both. The neighbor uses the ORFs you send as outbound filters when it sends routes to the Layer 3 switch. Likewise, the Layer 3 switch uses the ORFs it receives from the neighbor as outbound filters when sending routes to the neighbor.
- Reset the BGP4 neighbor session to send and receive ORFs.
- Perform these steps on the other device.

NOTE
If the Layer 3 switch has inbound filters, the filters are still processed even if equivalent filters have been sent as ORFs to the neighbor.

Enabling cooperative filtering

To configure cooperative filtering, enter commands such as the following.

```
Brocade(config)#ip prefix-list Routesfrom1234 deny 10.20.0.0/24
Brocade(config)#ip prefix-list Routesfrom1234 permit 0.0.0.0/0 le 32
Brocade(config)#router bgp
Brocade(config-bgp-router)#neighbor 10.2.3.4 prefix-list Routesfrom1234 in
Brocade(config-bgp-router)#neighbor 10.2.3.4 capability orf prefixlist send
```

The first two commands configure statements for the IP prefix list Routesfrom1234. The first command configures a statement that denies routes to 10.20.20./24. The second command configures a statement that permits all other routes. (Once you configure an IP prefix list statement, all routes not explicitly permitted by statements in the prefix list are denied.)

The next two commands change the CLI to the BGP4 configuration level, then apply the IP prefix list to neighbor 10.2.3.4. The last command enables the Layer 3 switch to send the IP prefix list as an ORF to neighbor 10.2.3.4. When the Layer 3 switch sends the IP prefix list to the neighbor, the neighbor filters out the 10.20.0.x routes from its updates to the Layer 3 switch. (This assumes that the neighbor also is configured for cooperative filtering.)

The *ip-addr* | *peer-group-name* parameter specifies the IP address of a neighbor or the name of a peer group of neighbors.

The **send** | **receive** parameter specifies the support you are enabling:

- **send** – The Layer 3 switch sends the IP prefix lists to the neighbor.
- **receive** – The Layer 3 switch accepts filters from the neighbor.

If you do not specify the capability, both capabilities are enabled.

The **prefixlist** parameter specifies the type of filter you want to send to the neighbor.

NOTE
The current release supports cooperative filtering only for filters configured using IP prefix lists.

Sending and receiving ORFs

Cooperative filtering affects neighbor sessions that start after the filtering is enabled, but do not affect sessions that are already established.

To activate cooperative filtering, reset the session with the neighbor. This is required because the cooperative filtering information is exchanged in Open messages during the start of a session.

To place a prefix-list change into effect after activating cooperative filtering, perform a soft reset of the neighbor session. A soft reset does not end the current session, but sends the prefix list to the neighbor in the next route refresh message.

NOTE
Make sure cooperative filtering is enabled on the Layer 3 switch and on the neighbor before you send the filters.

To reset a neighbor session and send ORFs to the neighbor, enter a command such as the following.

```
Brocade#clear ip bgp neighbor 10.2.3.4
```

This command resets the BGP4 session with neighbor 10.2.3.4 and sends the ORFs to the neighbor. If the neighbor sends ORFs to the Layer 3 switch, the Layer 3 switch accepts them if the send capability is enabled.

To perform a soft reset of a neighbor session and send ORFs to the neighbor, enter a command such as the following.

```
Brocade#clear ip bgp neighbor 10.2.3.4 soft in prefix-list
```

Syntax: **clear ip bgp neighbor** *ip-addr* [**soft in prefix-filter**]

If you use the **soft in prefix-filter** parameter, the Layer 3 switch sends the updated IP prefix list to the neighbor as part of its route refresh message to the neighbor.

NOTE
If the Layer 3 switch or the neighbor is not configured for cooperative filtering, the command sends a normal route refresh message.

Displaying cooperative filtering information

You can display the following cooperative filtering information:

- The cooperative filtering configuration on the Layer 3 switch.
- The ORFs received from neighbors.

To display the cooperative filtering configuration on the Layer 3 switch, enter a command such as the following. The line shown in bold type shows the cooperative filtering status.

```
Brocade#show ip bgp neighbors 10.10.10.1
1   IP Address: 10.10.10.1, AS: 65200 (IBGP), RouterID: 10.10.10.1
    State: ESTABLISHED, Time: 0h0m7s, KeepAliveTime: 60, HoldTime: 180
       RefreshCapability: Received
       CooperativeFilteringCapability: Received
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       0       1         0            1
       Received: 1       0       1         0            1
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                  Tx: ---        ---           Rx: ---        ---
    Last Connection Reset Reason:Unknown
    Notification Sent:     Unspecified
    Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
       Byte Sent:   110, Received: 110
       Local host:  10.10.10.2, Local  Port: 8138
       Remote host: 10.10.10.1, Remote Port: 179
       ISentSeq:        460  SendNext:        571  TotUnAck:          0
       TotSent:         111  ReTrans:           0  UnAckSeq:        571
       IRcvSeq:        7349  RcvNext:        7460  SendWnd:       16384
       TotalRcv:        111  DupliRcv:          0  RcvWnd:        16384
       SendQue:           0  RcvQue:            0  CngstWnd:       5325
```

Syntax: **show ip bgp neighbors** *ip-addr*

To display the ORFs received from a neighbor, enter a command such as the following.

```
Brocade#show ip bgp neighbors 10.10.10.1 received prefix-filter
ip prefix-list 10.10.10.1: 4 entries
     seq 5 permit 10.10.0.0/16 ge 18 le 28
     seq 10 permit 10.20.10.0/24
     seq 15 permit 10.30.0.0/8 le 32
     seq 20 permit 10.40.0.0/16 ge 18
```

Syntax: **show ip bgp neighbors** *ip-addr* **received prefix-filter**

Route flap dampening configuration

A “route flap” is the change in a route state, from up to down or down to up. When a route state changes, the state change causes changes in the route tables of the routers that support the route. Frequent changes in a route state can cause Internet instability and add processing overhead to the routers that support the route.

Route flap dampening is a mechanism that reduces the impact of route flap by changing a BGP4 router response to route state changes. When route flap dampening is configured, the Layer 3 switch suppresses unstable routes until the route state changes reduce enough to meet an acceptable degree of stability. The Brocade implementation of route flap dampening is based on RFC 2439.

Route flap dampening is disabled by default. You can enable the feature globally or on an individual route basis using route maps.

NOTE
The Layer 3 switch applies route flap dampening only to routes learned from EBGP neighbors.

The route flap dampening mechanism is based on penalties. When a route exceeds a configured penalty value, the Layer 3 switch stops using that route and also stops advertising it to other routers. The mechanism also allows a route penalties to reduce over time if the route stability improves. The route flap dampening mechanism uses the following parameters:

- Suppression threshold – Specifies the penalty value at which the Layer 3 switch stops using the route. Each time a route becomes unreachable or is withdrawn by a BGP4 UPDATE from a neighbor, the route receives a penalty of 1000. By default, when a route has a penalty value greater than 2000, the Layer 3 switch stops using the route. Thus, by default, if a route goes down more than twice, the Layer 3 switch stops using the route. You can set the suppression threshold to a value from 1 through 20000. The default is 2000.
- Half-life – Once a route has been assigned a penalty, the penalty decreases exponentially and decreases by half after the half-life period. The default half-life period is 15 minutes. The software reduces route penalties every five seconds. For example, if a route has a penalty of 2000 and does not receive any more penalties (it does not go down again) during the half-life, the penalty is reduced to 1000 after the half-life expires. You can configure the half-life to be from 1 through 45 minutes. The default is 15 minutes.
- Reuse threshold – Specifies the minimum penalty a route can have and still be suppressed by the Layer 3 switch. If the route's penalty falls below this value, the Layer 3 switch un-suppresses the route and can use it again. The software evaluates the dampened routes every ten seconds and un-suppresses the routes that have penalties below the reuse threshold. You can set the reuse threshold to a value from 1 through 20000. The default is 750.
- Maximum suppression time – Specifies the maximum number of minutes a route can be suppressed regardless of how unstable the route has been before this time. You can set the parameter to a value from
  1 through 20000 minutes. The default is four times the half-life. When the half-life value is set to its default (15 minutes), the maximum suppression time defaults to 60 minutes.

You can configure route flap dampening globally or for individual routes using route maps. If you configure route flap dampening parameters globally and also use route maps, the settings in the route maps override the global values.

Globally configuring route flap dampening

To enable route flap dampening using the default values, enter the following command.

```
Brocade(config-bgp-router)#dampening
```

Syntax: **dampening** [*half-life reuse suppress max-suppress-time*]

The *half-life* parameter specifies the number of minutes after which the route penalty becomes half its value. The route penalty allows routes that have remained stable for a while despite earlier instability to eventually become eligible for use again. The decay rate of the penalty is proportional to the value of the penalty. After the half-life expires, the penalty decays to half its value. Thus, a dampened route that is no longer unstable can eventually become eligible for use again. You can configure the half-life to be from 1 - 45 minutes. The default is 15 minutes.

The *reuse* parameter specifies how low a route penalty must become before the route becomes eligible for use again after being suppressed. You can set the reuse threshold to a value from 1 through 20000. The default is 750 (0.75, or three-fourths, of the penalty assessed for a one "flap").

The *suppress* parameter specifies how high a route penalty can become before the Layer 3 switch suppresses the route. You can set the suppression threshold to a value from 1 through 20000. The default is 2000 (two "flaps").

The *max-suppress-time* parameter specifies the maximum number of minutes that a route can be suppressed regardless of how unstable it is. You can set the maximum suppression time to a value from 1 through 20000 minutes. The default is four times the half-life setting. Thus, if you use the default half-life of 15 minutes, the maximum suppression time is 60 minutes.

The following example shows how to change the dampening parameters.

```
Brocade(config-bgp-router)#dampening 20 200 2500 40
```

This command changes the half-life to 20 minutes, the reuse threshold to 200, the suppression threshold to 2500, and the maximum number of minutes a route can be dampened to 40.

NOTE
To change any of the parameters, you must specify all the parameters with the command. If you want to leave some parameters unchanged, enter their default values.

Using a route map to configure route flap dampening for specific routes

Route maps enable you to fine tune route flap dampening parameters for individual routes. To configure route flap dampening parameters using route maps, configure BGP4 address filters for each route you want to set the dampening parameters for, then configure route map entries that set the dampening parameters for those routes. The following sections show examples.

To configure address filters and a route map for dampening specific routes, enter commands such as the following.

```
Brocade(config)#router bgp
Brocade(config-bgp-router)#address-filter 9 permit 10.157.22.0 255.255.255.0
255.255.255.0 255.255.255.0
Brocade(config-bgp-router)#address-filter 10 permit 10.157.23.0 255.255.255.0
255.255.255.0 255.255.255.0
Brocade(config-bgp-router)#exit
Brocade(config)#route-map DAMPENING_MAP permit 9
Brocade(config-routemap DAMPENING_MAP)#match address-filters 9
Brocade(config-routemap DAMPENING_MAP)#set dampening 10 200 2500 40
Brocade(config-routemap DAMPENING_MAP)#exit
Brocade(config)#route-map DAMPENING_MAP permit 10
Brocade(config-routemap DAMPENING_MAP)#match address-filters 10
Brocade(config-routemap DAMPENING_MAP)#set dampening 20 200 2500 60
Brocade(config-routemap DAMPENING_MAP)#router bgp
Brocade(config-bgp-router)#dampening route-map DAMPENING_MAP
```

The **address-filter** commands in this example configure two BGP4 address filters, for networks 10.157.22.0 and 10.157.23.0. The first route-map command creates an entry in a route map called "DAMPENING_MAP". Within this entry of the route map, the **match** command matches based on address filter 9, and the **set** command sets the dampening parameters for the route that matches. Thus, for BGP4 routes to 10.157.22.0, the Layer 3 switch uses the route map to set the dampening parameters. These parameters override the globally configured dampening parameters.

The commands for the second entry in the route map (instance 10 in this example) perform the same functions for route 10.157.23.0. Notice that the dampening parameters are different for each route.

Using a route map to configure route flap dampening for a specific neighbor

You can use a route map to configure route flap dampening for a specific neighbor by performing the following tasks:

- Configure an empty route map with no match or set statements. This route map does not specify particular routes for dampening but does allow you to enable dampening globally when you refer to this route map from within the BGP configuration level.
- Configure another route map that explicitly enables dampening. Use a set statement within the route map to enable dampening. When you associate this route map with a specific neighbor, the route map enables dampening for all routes associated with the neighbor. You also can use match statements within the route map to selectively perform dampening on some routes from the neighbor.

NOTE
You still need to configure the first route map to enable dampening globally. The second route map does not enable dampening by itself; it just applies dampening to a neighbor.

- Apply the route map to the neighbor.

To enable route flap dampening for a specific BGP4 neighbor, enter commands such as the following.

```
Brocade(config)#route-map DAMPENING_MAP_ENABLE permit 1
Brocade(config-routemap DAMPENING_MAP_ENABLE)#exit
Brocade(config)#route-map DAMPENING_MAP_NEIGHBOR_A permit 1
Brocade(config-routemap DAMPENING_MAP_NEIGHBOR_A)#set dampening
Brocade(config-routemap DAMPENING_MAP_NEIGHBOR_A)#exit
Brocade(config)#router bgp
Brocade(config-bgp-router)#dampening route-map DAMPENING_MAP_ENABLE
Brocade(config-bgp-router)#neighbor 10.10.10.1 route-map in
DAMPENING_MAP_NEIGHBOR_A
```

In this example, the first command globally enables route flap dampening. This route map does not contain any match or set statements. At the BGP configuration level, the **dampening route-map** command refers to the DAMPENING_MAP_ENABLE route map created by the first command, thus enabling dampening globally.

The third and fourth commands configure a second route map that explicitly enables dampening. Notice that the route map does not contain a match statement. The route map implicitly applies to all routes. Since the route map will be applied to a neighbor at the BGP configuration level, the route map will apply to all routes associated with the neighbor.

Although the second route map enables dampening, the first route map is still required. The second route map enables dampening for the neighbors to which the route map is applied. However, unless dampening is already enabled globally by the first route map, the second route map has no effect.

The last two commands apply the route maps. The **dampening route-map** command applies the first route map, which enables dampening globally. The **neighbor** command applies the second route map to neighbor 10.10.10.1. Since the second route map does not contain match statements for specific routes, the route map enables dampening for all routes received from the neighbor.

Removing route dampening from a route

You can un-suppress routes by removing route flap dampening from the routes. The Layer 3 switch allows you to un-suppress all routes at once or un-suppress individual routes.

To un-suppress all the suppressed routes, enter the following command at the Privileged EXEC level of the CLI.

```
Brocade#clear ip bgp damping
```

Syntax: **clear ip bgp damping** [*ip-addr ip-mask*]

The *ip-addr* parameter specifies a particular network.

The *ip-mask* parameter specifies the network mask.

To un-suppress a specific route, enter a command such as the following.

```
Brocade#clear ip bgp damping 10.157.22.0 255.255.255.0
```

This command un-suppresses only the routes for network 10.157.22.0/24.

Removing route dampening from neighbor routes suppressed due to aggregation

You can selectively unsuppress more-specific routes that have been suppressed due to aggregation, and allow the routes to be advertised to a specific neighbor or peer group.

Here is an example.

```
Brocade(config-bgp-router)#aggregate-address 10.1.0.0 255.255.0.0 summary-only
Brocade(config-bgp-router)#show ip bgp route 10.1.0.0/16 longer
Number of BGP Routes matching display condition : 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.1.0.0/16       0.0.0.0                    101        32768  BAL
         AS_PATH:
2      10.1.44.0/24      10.2.0.1        1          101        32768  BLS
         AS_PATH:
```

The **aggregate-address** command configures an aggregate address. The **summary-only p**arameter prevents the Layer 3 switch from advertising more specific routes contained within the aggregate route. The **show ip bgp route** command shows that the more specific routes aggregated into 10.1.0.0/16 have been suppressed. In this case, the route to 10.1.44.0/24 has been suppressed. The following command indicates that the route is not being advertised to the Layer 3 switch BGP4 neighbors.

```
Brocade#show ip bgp route 10.1.44.0/24
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.1.44.0/24      10.2.0.1        1          101        32768  BLS
         AS_PATH:
       Route is not advertised to any peers
```

If you want to override the **summary-only** parameter and allow a specific route to be advertised to a neighbor, enter commands such as the following.

```
Brocade(config)#ip prefix-list Unsuppress1 permit 10.1.44.0/24
Brocade(config)#route-map RouteMap1 permit 1
Brocade(config-routemap RouteMap1)#match ip prefix-list Unsuppress1
Brocade(config-routemap RouteMap1)#exit
Brocade(config)#router bgp
Brocade(config-bgp-router)#neighbor 10.1.0.2 unsuppress-map RouteMap1
Brocade(config-bgp-router)#clear ip bgp neighbor 10.1.0.2 soft-out
```

The **ip prefix-list** command configures an IP prefix list for network 10.1.44.0/24, which is the route you want to unsuppress. The next two commands configure a route map that uses the prefix list as input. The **neighbor** command enables the Layer 3 switch to advertise the routes specified in the route map to neighbor 10.1.0.2. The **clear** command performs a soft reset of the session with the neighbor so that the Layer 3 switch can advertise the unsuppressed route.

Syntax: [**no**] **neighbor** *ip-addr* | *peer-group-name* **unsuppress-map** *map-name*

The following command verifies that the route has been unsuppressed.

```
Brocade#show ip bgp route 10.1.44.0/24
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.1.44.0/24      10.2.0.1        1          101        32768  BLS
         AS_PATH:
       Route is advertised to 1 peers:
        10.1.0.2(4)
```

Displaying and clearing route flap dampening statistics

The software provides many options for displaying and clearing route flap statistics. To display the statistics, use either of the following methods.

Displaying route flap dampening statistics

To display route dampening statistics or all the dampened routes, enter the **show ip bgp flap-statistics** command at any level of the CLI.

```
Brocade#show ip bgp flap-statistics
Total number of flapping routes: 414
    Status Code  >:best d:damped h:history *:valid
    Network            From            Flaps Since    Reuse    Path
h>  192.50.206.0/23    10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  192.168.192.0/20   10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.168.165.0/24   10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.168.208.0/23   10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  192.168.0.0/16     10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  192.168.220.0/24   10.90.213.77   1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

Syntax: **show ip bgp flap-statistics** [**regular-expression** *regular-expression* | *address mask* [**longer-prefixes**] | **neighbor** *ip-addr*]

The **regular-expression** *regular-expression* parameter is a regular expression. The regular expressions are the same ones supported for BGP4 AS-path filters. Refer to "Using regular expressions to filter" on page 336.

The *address mask* parameter specifies a particular route. If you also use the optional **longer-prefixes** parameter, then all statistics for routes that match the specified route or have a longer prefix than the specified route are displayed. For example, if you specify **10.157.0.0 longer**, then all routes with the prefix 10.157. or that have a longer prefix (such as 10.157.22.) are displayed.

The **neighbor** *ip-addr* parameter displays route flap dampening statistics only for routes learned from the specified neighbor. You also can display route flap statistics for routes learned from a neighbor by entering the following command: **show ip bgp neighbors** *ip-addr* **flap-statistics.**

Table 63 shows the field definitions for the display output.

TABLE 63 **Route flap dampening statistics**

Field	Description
Total number of flapping routes	Total number of routes in the Layer 3 switch BGP4 route table that have changed state and thus have been marked as flapping routes.
Status code	Indicates the dampening status of the route, which can be one of the following: • > – This is the best route among those in the BGP4 route table to the route destination. • d – This route is currently dampened, and thus unusable. • h – The route has a history of flapping and is unreachable now. • * – The route has a history of flapping but is currently usable.
Network	The destination network of the route.
From	The neighbor that sent the route to the Layer 3 switch.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time since the first flap of this route.
Reuse	The amount of time remaining until this route will be un-suppressed and thus be usable again.
Path	Shows the AS-path information for the route.

You also can display all the dampened routes by entering the **show ip bgp dampened-paths** command.

Clearing route flap dampening statistics

To clear route flap dampening statistics, use the following CLI method.

NOTE
Clearing the dampening statistics for a route does not change the dampening status of the route.

To clear all the route dampening statistics, enter the following command at any level of the CLI.

```
Brocade#clear ip bgp flap-statistics
```

Syntax: **clear ip bgp flap-statistics** [**regular-expression** *regular-expression* | *address mask* | **neighbor** *ip-addr*]

The parameters are the same as those for the **show ip bgp flap-statistics** command (except the **longer-prefixes** option is not supported). Refer to "Displaying route flap dampening statistics" on page 359.

NOTE
The **clear ip bgp damping** command not only clears statistics but also un-suppresses the routes. Refer to "Displaying route flap dampening statistics" on page 359.

Generating traps for BGP

You can enable and disable SNMP traps for BGP. BGP traps are enabled by default.

To enable BGP traps after they have been disabled, enter the following command.

```
Brocade(config)#snmp-server enable traps bgp
```

Syntax: [**no**] **snmp-server enable traps bgp**

Use the **no** form of the command to disable BGP traps.

Displaying BGP4 information

You can display the following configuration information and statistics for the BGP4 protocol on the router:

- Summary BGP4 configuration information for the router
- Active BGP4 configuration information (the BGP4 information in the running-config)
- CPU utilization statistics
- Neighbor information
- Peer-group information
- Information about the paths from which BGP4 selects routes
- Summary BGP4 route information
- The router BGP4 route table
- Route flap dampening statistics
- Active route maps (the route map configuration information in the running-config)
- BGP4 graceful restart neighbor information

Displaying summary BGP4 information

You can display the local AS number, the maximum number of routes and neighbors supported, and some BGP4 statistics.

To view summary BGP4 information for the router, enter the **show ip bgp summary** command at any CLI prompt.

```
Brocade#show ip bgp summary
  BGP4 Summary
  Router ID: 10.0.0.1   Local AS Number : 4
  Confederation Identifier : not configured
  Confederation Peers: 4 5
  Maximum Number of Paths Supported for Load Sharing : 1
  Number of Neighbors Configured : 11
  Number of Routes Installed : 2
  Number of Routes Advertising to All Neighbors : 8
  Number of Attribute Entries Installed : 6
  Neighbor Address  AS#  State   Time     Rt:Accepted Filtered Sent   ToSend
  10.2.3.4           200   ADMDN   0h44m56s    0        0        0     2
  10.0.0.2          5     ADMDN   0h44m56s    0        0        0      0
  10.1.0.2          5     ESTAB   0h44m56s    1        11       0      0
  10.2.0.2          5     ESTAB   0h44m55s    1        0        0      0
  10.3.0.2          5     ADMDN   0h25m28s    0        0        0      0
  10.4.0.2          5     ADMDN   0h25m31s    0        0        0      0
  10.5.0.2          5     CONN    0h 0m 8s    0        0        0      0
  10.7.0.2          5     ADMDN   0h44m56s    0        0        0      0
  10.10.0.1         4     ADMDN   0h44m56s    0        0        0      2
  10.10.0.1         4     ADMDN   0h44m56s    0        0        0      2
  10.150.150.150   0     ADMDN   0h44m56s    0        0        0      2
```

Table 64 lists the field definitions for the command output.

TABLE 64 BGP4 summary information

Field	Description
Router ID	The Layer 3 switch router ID.
Local AS Number	The BGP4 AS number the router is in.
Confederation Identifier	The AS number of the confederation the Layer 3 switch is in.
Confederation Peers	The numbers of the local autonomous systems contained in the confederation. This list matches the confederation peer list you configure on the Layer 3 switch.
Maximum Number of Paths Supported for Load Sharing	The maximum number of route paths across which the device can balance traffic to the same destination. The feature is enabled by default but the default number of paths is 1. You can increase the number from 2 through 4 paths. Refer to "Changing the maximum number of paths for BGP4 load sharing" on page 305.
Number of Neighbors Configured	The number of BGP4 neighbors configured on this Layer 3 switch.
Number of Routes Installed	The number of BGP4 routes in the router BGP4 route table. To display the BGP4 route table, refer to "Displaying the BGP4 route table" on page 380.
Number of Routes Advertising to All Neighbors	The total of the RtSent and RtToSend columns for all neighbors.
Number of Attribute Entries Installed	The number of BGP4 route-attribute entries in the router route-attributes table. To display the route-attribute table, refer to "Displaying BGP4 route-attribute entries" on page 386.
Neighbor Address	The IP addresses of this router BGP4 neighbors.
AS#	The AS number.

TABLE 64 BGP4 summary information (Continued)

Field	Description
State	The state of this router neighbor session with each neighbor. The states are from this router perspective of the session, not the neighbor perspective. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each router: • IDLE – The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • ADMND – The neighbor has been administratively shut down. Refer to "Administratively shutting down a session with a BGP4 neighbor" on page 302. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • CONNECT – BGP4 is waiting for the connection process for the TCP neighbor session to be completed. • ACTIVE – BGP4 is waiting for a TCP connection from the neighbor. NOTE: If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection. • OPEN SENT – BGP4 is waiting for an Open message from the neighbor. • OPEN CONFIRM – BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the router receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle. • ESTABLISHED – BGP4 is ready to exchange UPDATE packets with the neighbor. If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed. NOTE: If you display information for the neighbor using the **show ip bgp neighbors** *ip-addr* command, the TCP receiver queue value will be greater than 0. Operational States: Additional information regarding the BGP operational states described above may be added as follows: • (+) – is displayed if there is more BGP4 data in the TCP receiver queue. **Note:** If you display information for the neighbor using the show ip bgp neighbors *ip-addr* command, the TCP receiver queue value will be greater than 0. • (-) – indicates that the session has gone down and the software is clearing or removing routes. • (*) – indicates that the inbound or outbound policy is being updated for the peer. • (s) – indicates that the peer has negotiated restart, and the session is in a stale state. • (r) – indicates that the peer is restarting the BGP4 connection, through restart. • (^) – on the standby MP indicates that the peer is in the ESTABLISHED state and has received restart capability (in the primary MP). • (<) – indicates that the device is waiting to receive the "End of RIB" message the peer.
Time	The time that has passed since the state last changed.
Accepted	The number of routes received from the neighbor that this router installed in the BGP4 route table. Usually, this number is lower than the RoutesRcvd number. The difference indicates that this router filtered out some of the routes received in the UPDATE messages.

TABLE 64 BGP4 summary information (Continued)

Field	Description
Filtered	The routes or prefixes that have been filtered out: • If soft reconfiguration is enabled, this field shows how many routes were filtered out (not placed in the BGP4 route table) but retained in memory. • If soft reconfiguration is not enabled, this field shows the number of BGP4 routes that have been filtered out.
Sent	The number of BGP4 routes that the Layer 3 switch has sent to the neighbor.
ToSend	The number of routes the Layer 3 switch has queued to send to this neighbor.

Displaying the active BGP4 configuration

To view the active BGP4 configuration information contained in the running-config without displaying the entire running-config, use the following CLI method.

To display the device active BGP4 configuration, enter the **show ip bgp config** command at any level of the CLI.

```
Brocade#show ip bgp config
Current BGP configuration:
router bgp
 address-filter  1 deny  any   any
 as-path-filter  1 permit ^65001$
 local-as 65002
 maximum-paths 4
 neighbor pg1 peer-group
 neighbor pg1 remote-as 65001
 neighbor pg1 description "Brocade group 1"
 neighbor pg1 distribute-list out 1
 neighbor 192.168.100.1 peer-group pg1
 neighbor 192.168.101.1 peer-group pg1
 neighbor 192.168.102.1 peer-group pg1
 neighbor 192.168.201.1 remote-as 65101
 neighbor 192.168.201.1 shutdown
 neighbor 192.168.220.3 remote-as 65432
 network 10.1.1.0 255.255.255.0
 network 10.2.2.0 255.255.255.0
 redistribute connected
```

Syntax: **show ip bgp config**

Displaying CPU utilization statistics

You can display CPU utilization statistics for BGP4 and other IP protocols.

To display CPU utilization statistics for BGP4 for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the **show process cpu** command at any level of the CLI.

```
Brocade#show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22              9
BGP              0.04      0.06      0.08      0.14             13
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00              0
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example.

```
Brocade#show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00              0
BGP              0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.00      0.00      0.00              1
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following.

```
Brocade#show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00       0
BGP              0.00       0
GVRP             0.00       0
ICMP             0.01       1
IP               0.00       0
OSPF             0.00       0
RIP              0.00       0
STP              0.01       0
VRRP             0.00       0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: **show process cpu** [*num*]

The *num* parameter specifies the number of seconds and can be from 1 through 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying summary neighbor information

To display summary neighbor information, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 routes-summary
1   IP Address: 192.168.4.211
Routes Accepted/Installed:1,  Filtered/Kept:11,  Filtered:11
   Routes Selected as BEST Routes:1
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0

NLRIs Received in Update Message:24,  Withdraws:0 (0),  Replacements:1
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0
```

Syntax: **show ip bgp neighbors** [*ip-addr*] | [**routes-summary**]

Table 65 lists the field definitions for the command output.

TABLE 65 BGP4 route summary information for a neighbor

Field	Description
IP Address	The IP address of the neighbor
Routes Received	How many routes the Layer 3 switch has received from the neighbor during the current BGP4 session: • Accepted/Installed – Indicates how many of the received routes the Layer 3 switch accepted and installed in the BGP4 route table. • Filtered/Kept – Indicates how many routes were filtered out, but were nonetheless retained in memory for use by the soft reconfiguration feature. • Filtered – Indicates how many of the received routes were filtered out.
Routes Selected as BEST Routes	The number of routes that the Layer 3 switch selected as the best routes to their destinations.
BEST Routes not Installed in IP Forwarding Table	The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the Layer 3 switch does not have a valid RIP, OSPF, or static route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.

TABLE 65 BGP4 route summary information for a neighbor (Continued)

Field	Description
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages: • Withdraws – The number of withdrawn routes the Layer 3 switch has received. • Replacements – The number of replacement routes the Layer 3 switch has received.
NLRIs Discarded due to	Indicates the number of times the Layer 3 switch discarded an NLRI for the neighbor due to the following reasons: • Maximum Prefix Limit – The Layer 3 switch configured maximum prefix amount had been reached. • AS Loop – An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number. • Invalid Nexthop – The next hop value was not acceptable. • Duplicated Originator_ID – The originator ID was the same as the local router ID. • Cluster_ID – The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured.
Routes Advertised	The number of routes the Layer 3 switch has advertised to this neighbor: • To be Sent – The number of routes the Layer 3 switch has queued to send to this neighbor. • To be Withdrawn – The number of NLRIs for withdrawing routes the Layer 3 switch has queued up to send to this neighbor in UPDATE messages.
NLRIs Sent in Update Message	The number of NLRIs for new routes the Layer 3 switch has sent to this neighbor in UPDATE messages: • Withdraws – The number of routes the Layer 3 switch has sent to the neighbor to withdraw. • Replacements – The number of routes the Layer 3 switch has sent to the neighbor to replace routes the neighbor already has.
Peer Out of Memory Count for	Statistics for the times the Layer 3 switch has run out of BGP4 memory for the neighbor during the current BGP4 session: • Receiving Update Messages – The number of times UPDATE messages were discarded because there was no memory for attribute entries. • Accepting Routes(NLRI) – The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count. • Attributes – The number of times there was no memory for BGP4 attribute entries. • Outbound Routes(RIB-out) – The number of times there was no memory to place a “best” route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.

Displaying BGP4 neighbor information

To view BGP4 neighbor information including the values for all the configured parameters, enter the following command.

NOTE
The display shows all the configured parameters for the neighbor. Only the parameters that have values different from their defaults are shown.

```
Brocade#show ip bgp neighbors 10.4.0.2
1   IP Address: 10.4.0.2, AS: 5 (EBGP), RouterID: 10.10.0.1
       Description: neighbor 10.4.0.2
    State: ESTABLISHED, Time: 0h1m0s, KeepAliveTime: 0, HoldTime: 0
       PeerGroup: pg1
       Multihop-EBGP: yes, ttl: 1
       RouteReflectorClient: yes
       SendCommunity: yes
       NextHopSelf: yes
       DefaultOriginate: yes (default sent)
       MaximumPrefixLimit: 90000
       RemovePrivateAs: : yes
       RefreshCapability: Received
    Route Filter Policies:
       Distribute-list: (out) 20
       Filter-list: (in) 30
       Prefix-list: (in) pf1
       Route-map: (in) setnp1  (out) setnp2
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       1       1         0            0
       Received: 1       8       1         0            0
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                  Tx: 0h0m59s    ---           Rx: 0h0m59s    ---
    Last Connection Reset Reason:Unknown
      Notification Sent:     Unspecified
      Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
       Local host:  10.4.0.1, Local  Port: 179
       Remote host: 10.4.0.2, Remote Port: 8053
       ISentSeq:   52837276  SendNext:   52837392  TotUnAck:          0
       TotSent:         116  ReTrans:          0  UnAckSeq:   52837392
       IRcvSeq:  2155052043  RcvNext:  2155052536  SendWnd:       16384
       TotalRcv:        493  DupliRcv:         0  RcvWnd:        16384
       SendQue:           0  RcvQue:           0  CngstWnd:       1460
```

This example shows how to display information for a specific neighbor, by specifying the neighbor IP address with the command. None of the other display options are used; thus, all of the information is displayed for the neighbor. The number in the far left column indicates the neighbor for which information is displayed. When you list information for multiple neighbors, this number makes the display easier to read.

The TCP statistics at the end of the display show status for the TCP session with the neighbor. Most of the fields show information stored in the Layer 3 switch Transmission Control Block (TCB) for the TCP session between the Layer 3 switch and its neighbor. These fields are described in detail in section 3.2 of RFC 793, “Transmission Control Protocol Functional Specification”.

Syntax: **show ip bgp neighbors** [*ip-addr* [**advertised-routes** [**detail** [*ip-addr*[/*mask-bits*]]]] | [**attribute-entries** [**detail**]] | [**flap-statistics**] | [**last-packet-with-error**] | [**received prefix-filter**] |
[**received-routes**] | [**routes** [**best**] | [**detail** [**best**] | [**not-installed-best**] | [**unreachable**]] |
[**rib-out-routes** [*ip-addr/mask-bits* | *ip-addr net-mask* | **detail**]] | [**routes-summary**]]

The *ip-addr* option lets you narrow the scope of the command to a specific neighbor.

The **advertised-routes** option displays only the routes that the Layer 3 switch has advertised to the neighbor during the current BGP4 neighbor session.

The **attribute-entries** option shows the attribute-entries associated with routes received from the neighbor.

The **flap-statistics** option shows the route flap statistics for routes received from or sent to the neighbor.

The **last-packet-with-error** option displays the last packet from the neighbor that contained an error. The packet's contents are displayed in decoded (human-readable) format.

The **received prefix-filter** option shows the Outbound Route Filters (ORFs) received from the neighbor. This option applies to cooperative route filtering.

The **received-routes** option lists all the route information received in route updates from the neighbor since the soft reconfiguration feature was enabled. Refer to "Using soft reconfiguration" on page 391.

The **routes** option lists the routes received in UPDATE messages from the neighbor. You can specify the following additional options:

- **best** – Displays the routes received from the neighbor that the Layer 3 switch selected as the best routes to their destinations.
- **not-installed-best** – Displays the routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
- **unreachable** – Displays the routes that are unreachable because the Layer 3 switch does not have a valid RIP, OSPF, or static route to the next hop.
- **detail** – Displays detailed information for the specified routes. You can refine your information request by also specifying one of the options above (**best, not-installed-best,** or **unreachable**).

The **rib-out-routes** option lists the route information base (RIB) for outbound routes. You can display all the routes or specify a network address.

The **routes-summary** option displays a summary of the following information:

- Number of routes received from the neighbor
- Number of routes accepted by this Layer 3 switch from the neighbor
- Number of routes this Layer 3 switch filtered out of the UPDATES received from the neighbor and did not accept
- Number of routes advertised to the neighbor
- Number of attribute entries associated with routes received from or advertised to the neighbor.

Table 66 lists the field definitions for the command output.

TABLE 66 BGP4 neighbor information

Field	Description
IP Address	The IP address of the neighbor.
AS	The AS the neighbor is in.
EBGP/IBGP	Whether the neighbor session is an IBGP session, an EBGP session, or a confederation EBGP session: • EBGP – The neighbor is in another AS. • EBGP_Confed – The neighbor is a member of another sub-AS in the same confederation. • IBGP – The neighbor is in the same AS.

TABLE 66 BGP4 neighbor information (Continued)

Field	Description
RouterID	The neighbor router ID.
Description	The description you gave the neighbor when you configured it on the Layer 3 switch.
State	The state of the router session with the neighbor. The states are from this router perspective of the session, not the neighbor perspective. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each router: • IDLE – The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • ADMND – The neighbor has been administratively shut down. Refer to "Administratively shutting down a session with a BGP4 neighbor" on page 302. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes. • CONNECT – BGP4 is waiting for the connection process for the TCP neighbor session to be completed. • ACTIVE – BGP4 is waiting for a TCP connection from the neighbor. NOTE: If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection. • OPEN SENT – BGP4 is waiting for an Open message from the neighbor. • OPEN CONFIRM – BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the router receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle. • ESTABLISHED – BGP4 is ready to exchange UPDATE messages with the neighbor. If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed. NOTE: If you display information for the neighbor using the **show ip bgp neighbors** *ip-addr* command, the TCP receiver queue value will be greater than 0.
Time	The amount of time this session has been in its current state.
KeepAliveTime	The keep alive time, which specifies how often this router sends keep alive messages to the neighbor. Refer to "Changing the Keep Alive Time and Hold Time" on page 304.
HoldTime	The hold time, which specifies how many seconds the router will wait for a KEEPALIVE or UPDATE message from a BGP4 neighbor before deciding that the neighbor is dead. Refer to "Changing the Keep Alive Time and Hold Time" on page 304.
PeerGroup	The name of the peer group the neighbor is in, if applicable.
Multihop-EBGP	Whether this option is enabled for the neighbor.
RouteReflectorClient	Whether this option is enabled for the neighbor.
SendCommunity	Whether this option is enabled for the neighbor.
NextHopSelf	Whether this option is enabled for the neighbor.
DefaultOriginate	Whether this option is enabled for the neighbor.
MaximumPrefixLimit	Lists the maximum number of prefixes the Layer 3 switch will accept from this neighbor.

TABLE 66 BGP4 neighbor information (Continued)

Field	Description
RemovePrivateAs	Whether this option is enabled for the neighbor.
RefreshCapability	Whether this Layer 3 switch has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability.
CooperativeFilteringCapability	Whether the neighbor is enabled for cooperative route filtering.
Distribute-list	Lists the distribute list parameters, if configured.
Filter-list	Lists the filter list parameters, if configured.
Prefix-list	Lists the prefix list parameters, if configured.
Route-map	Lists the route map parameters, if configured.
Messages Sent	The number of messages this router has sent to the neighbor. The display shows statistics for the following message types: • Open • Update • KeepAlive • Notification • Refresh-Req
Messages Received	The number of messages this router has received from the neighbor. The message types are the same as for the Message Sent field.
Last Update Time	Lists the last time updates were sent and received for the following: • NLRIs • Withdraws

TABLE 66 BGP4 neighbor information (Continued)

Field	Description
Last Connection Reset Reason	The reason the previous session with this neighbor ended. The reason can be one of the following. Reasons described in the BGP specifications: • Message Header Error • Connection Not Synchronized • Bad Message Length • Bad Message Type • OPEN Message Error • Unsupported Version Number • Bad Peer AS Number • Bad BGP Identifier • Unsupported Optional Parameter • Authentication Failure • Unacceptable Hold Time • Unsupported Capability • UPDATE Message Error • Malformed Attribute List • Unrecognized Well-known Attribute • Missing Well-known Attribute • Attribute Flags Error • Attribute Length Error • Invalid ORIGIN Attribute • Invalid NEXT_HOP Attribute • Optional Attribute Error • Invalid Network Field • Malformed AS_PATH • Hold Timer Expired • Finite State Machine Error • Rcv Notification
Last Connection Reset Reason (cont.)	Reasons specific to the Brocade implementation: • Reset All Peer Sessions • User Reset Peer Session • Port State Down • Peer Removed • Peer Shutdown • Peer AS Number Change • Peer AS Confederation Change • TCP Connection KeepAlive Timeout • TCP Connection Closed by Remote • TCP Data Stream Error Detected

TABLE 66 BGP4 neighbor information (Continued)

Field	Description
Notification Sent	If the router receives a NOTIFICATION message from the neighbor, the message contains an error code corresponding to one of the following errors. Some errors have subcodes that clarify the reason for the error. Where applicable, the subcode messages are listed underneath the error code messages. Message Header Error: • Connection Not Synchronized • Bad Message Length • Bad Message Type • Unspecified Open Message Error: • Unsupported Version • Bad Peer As • Bad BGP Identifier • Unsupported Optional Parameter • Authentication Failure • Unacceptable Hold Time • Unspecified Update Message Error: • Malformed Attribute List • Unrecognized Attribute • Missing Attribute • Attribute Flag Error • Attribute Length Error • Invalid Origin Attribute • Invalid NextHop Attribute • Optional Attribute Error • Invalid Network Field • Malformed AS Path • Unspecified Hold Timer Expired Finite State Machine Error Cease Unspecified
Notification Received	See above.

TABLE 66 BGP4 neighbor information (Continued)

Field	Description
TCP Connection state	The state of the connection with the neighbor. The connection can have one of the following states: • LISTEN – Waiting for a connection request. • SYN-SENT – Waiting for a matching connection request after having sent a connection request. • SYN-RECEIVED – Waiting for a confirming connection request acknowledgment after having both received and sent a connection request. • ESTABLISHED – Data can be sent and received over the connection. This is the normal operational state of the connection. • FIN-WAIT-1 – Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent. • FIN-WAIT-2 – Waiting for a connection termination request from the remote TCP. • CLOSE-WAIT – Waiting for a connection termination request from the local user. • CLOSING – Waiting for a connection termination request acknowledgment from the remote TCP. • LAST-ACK – Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request). • TIME-WAIT – Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request. • CLOSED – There is no connection state.
Byte Sent	The number of bytes sent.
Byte Received	The number of bytes received.
Local host	The IP address of the Layer 3 switch.
Local port	The TCP port the Layer 3 switch is using for the BGP4 TCP session with the neighbor.
Remote host	The IP address of the neighbor.
Remote port	The TCP port the neighbor is using for the BGP4 TCP session with the Layer 3 switch.
ISentSeq	The initial send sequence number for the session.
SendNext	The next sequence number to be sent.
TotUnAck	The number of sequence numbers sent by the Layer 3 switch that have not been acknowledged by the neighbor.
TotSent	The number of sequence numbers sent to the neighbor.
ReTrans	The number of sequence numbers that the Layer 3 switch retransmitted because they were not acknowledged.
UnAckSeq	The current acknowledged sequence number.
IRcvSeq	The initial receive sequence number for the session.
RcvNext	The next sequence number expected from the neighbor.
SendWnd	The size of the send window.
TotalRcv	The number of sequence numbers received from the neighbor.
DupliRcv	The number of duplicate sequence numbers received from the neighbor.

TABLE 66 **BGP4 neighbor information (Continued)**

Field	Description
RcvWnd	The size of the receive window.
SendQue	The number of sequence numbers in the send queue.
RcvQue	The number of sequence numbers in the receive queue.
CngstWnd	The number of times the window has changed.

Displaying route information for a neighbor

You can display routes based on the following criteria:

- A summary of the routes for a specific neighbor.
- The routes received from the neighbor that the Layer 3 switch selected as the best routes to their destinations.
- The routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
- The routes that are unreachable because the Layer 3 switch does not have a valid RIP, OSPF, or static route to the next hop.
- Routes for a specific network advertised by the Layer 3 switch to the neighbor.
- The Routing Information Base (RIB) for a specific network advertised to the neighbor. You can display the RIB regardless of whether the Layer 3 switch has already sent it to the neighbor.

To display route information for a neighbor, use the following CLI methods.

Displaying summary route information

To display summary route information, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 10.1.0.2 routes-summary
1   IP Address: 10.1.0.2
Routes Accepted/Installed:1,  Filtered/Kept:11,  Filtered:11
   Routes Selected as BEST Routes:1
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0

NLRIs Received in Update Message:24,  Withdraws:0 (0),  Replacements:1
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0
```

Table 67 lists the field definitions for the command output.

TABLE 67 BGP4 route summary information for a neighbor

Field	Description
Routes Received	How many routes the Layer 3 switch has received from the neighbor during the current BGP4 session: • **Accepted/Installed** – Indicates how many of the received routes the Layer 3 switch accepted and installed in the BGP4 route table. • **Filtered** – Indicates how many of the received routes the Layer 3 switch did not accept or install because they were denied by filters on the Layer 3 switch.
Routes Selected as BEST Routes	The number of routes that the Layer 3 switch selected as the best routes to their destinations.
BEST Routes not Installed in IP Forwarding Table	The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the Layer 3 switch does not have a valid RIP, OSPF, or static route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages: • **Withdraws** – The number of withdrawn routes the Layer 3 switch has received. • **Replacements** – The number of replacement routes the Layer 3 switch has received.
NLRIs Discarded due to	Indicates the number of times the Layer 3 switch discarded an NLRI for the neighbor due to the following reasons: • **Maximum Prefix Limit** – The Layer 3 switch configured maximum prefix amount had been reached. • **AS Loop** – An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number. • **Invalid Nexthop** – The next hop value was not acceptable. • **Duplicated Originator_ID** – The originator ID was the same as the local router ID. • **Cluster_ID** – The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured.
Routes Advertised	The number of routes the Layer 3 switch has advertised to this neighbor: • **To be Sent** – The number of routes the Layer 3 switch has queued to send to this neighbor. • **To be Withdrawn** – The number of NLRIs for withdrawing routes the Layer 3 switch has queued up to send to this neighbor in UPDATE messages.

TABLE 67 BGP4 route summary information for a neighbor (Continued)

Field	Description
NLRIs Sent in Update Message	The number of NLRIs for new routes the Layer 3 switch has sent to this neighbor in UPDATE messages: • **Withdraws** – The number of routes the Layer 3 switch has sent to the neighbor to withdraw. • **Replacements** – The number of routes the Layer 3 switch has sent to the neighbor to replace routes the neighbor already has.
Peer Out of Memory Count for	Statistics for the times the Layer 3 switch has run out of BGP4 memory for the neighbor during the current BGP4 session: • **Receiving Update Messages** – The number of times UPDATE messages were discarded because there was no memory for attribute entries. • **Accepting Routes(NLRI)** – The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count. • **Attributes** – The number of times there was no memory for BGP4 attribute entries. • **Outbound Routes(RIB-out)** – The number of times there was no memory to place a “best” route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.

Displaying advertised routes

To display the routes the Layer 3 switch has advertised to a specific neighbor for a specific network, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 advertised-routes
       There are 2 routes advertised to neighbor 192.168.4.211
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Network        Next Hop        Metric     LocPrf     Weight     Status
1      10.0.0.0/24   192.168.2.102   12                     32768      BL
2      10.1.1.0/24   192.168.2.102   0                      32768    BL
```

You also can enter a specific route, as in the following example.

```
Brocade#show ip bgp neighbors 192.168.4.211 advertised 10.1.1.0/24
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Network        Next Hop        Metric     LocPrf     Weight     Status
1      10.1.1.0/24   192.168.2.102   0                      32768      BL
```

Syntax: **show ip bgp neighbors** *ip-addr* **advertised-routes** [*ip-addr/prefix*]

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying the best routes

To display the routes received from a specific neighbor that are the “best” routes to their destinations, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 routes best
```

Syntax: **show ip bgp neighbors** *ip-addr* **routes best**

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying the best routes that were nonetheless not installed in the IP route table

To display the BGP4 routes received from a specific neighbor that are the “best” routes to their destinations but are not installed in the Layer 3 switch IP route table, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 routes not-installed-best
```

Each of the displayed routes is a valid path to its destination, but the Layer 3 switch received another path from a different source (such as OSPF, RIP, or a static route) that has a lower administrative distance. The Layer 3 switch always selects the path with the lowest administrative distance to install in the IP route table.

Syntax: **show ip bgp neighbors** *ip-addr* **routes not-installed-best**

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying the routes whose destinations are unreachable

To display BGP4 routes whose destinations are unreachable using any of the BGP4 paths in the BGP4 route table, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 routes unreachable
```

Syntax: **show ip bgp neighbors** *ip-addr* **routes unreachable**

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying the Adj-RIB-Out for a neighbor

To display the Layer 3 switch current BGP4 Routing Information Base (Adj-RIB-Out) for a specific neighbor and a specific destination network, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.211 rib-out-routes 192.168.1.0/24
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.1.1.0/24       0.0.0.0         0          101        32768  BL
```

The Adj-RIB-Out contains the routes that the Layer 3 switch either has most recently sent to the neighbor or is about to send to the neighbor.

Syntax: **show ip bgp neighbors** *ip-addr* **rib-out-routes** [*ip-addr/prefix*]

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying peer group information

You can display configuration information for peer groups.

To display peer-group information, enter a command such as the following at the Privileged EXEC level of the CLI.

```
Brocade#show ip bgp peer-group pg1
1   BGP peer-group is pg
    Description: peer group abc
       SendCommunity: yes
       NextHopSelf: yes
       DefaultOriginate: yes
    Members:
       IP Address: 192.168.10.10, AS: 65111
```

Syntax: **show ip bgp peer-group** [*peer-group-name*]

Only the parameters that have values different from their defaults are listed.

Displaying summary route information

To display summary statistics for all the routes in the Layer 3 switch BGP4 route table, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp routes summary
  Total number of BGP routes (NLRIs) Installed     : 20
  Distinct BGP destination networks                : 20
  Filtered BGP routes for soft reconfig            : 100178
  Routes originated by this router                 : 2
  Routes selected as BEST routes                   : 19
  BEST routes not installed in IP forwarding table : 1
  Unreachable routes (no IGP route for NEXTHOP)    : 1
  IBGP routes selected as best routes              : 0
  EBGP routes selected as best routes              : 17
```

Syntax: **show ip bgp routes summary**

Table 68 lists the field definitions for the command output.

TABLE 68 BGP4 summary route information

Field	Description
Total number of BGP routes (NLRIs) Installed	The number of BGP4 routes the Layer 3 switch has installed in the BGP4 route table.
Distinct BGP destination networks	The number of destination networks the installed routes represent. The BGP4 route table can have multiple routes to the same network.
Filtered BGP routes for soft reconfig	The number of route updates received from soft-reconfigured neighbors or peer groups that have been filtered out but retained. For information about soft reconfiguration, refer to "Using soft reconfiguration" on page 391.
Routes originated by this router	The number of routes in the BGP4 route table that this Layer 3 switch originated.
Routes selected as BEST routes	The number of routes in the BGP4 route table that this Layer 3 switch has selected as the best routes to the destinations.
BEST routes not installed in IP forwarding table	The number of BGP4 routes that are the best BGP4 routes to their destinations but were not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable routes (no IGP route for NEXTHOP)	The number of routes in the BGP4 route table whose destinations are unreachable because the next hop is unreachable.

TABLE 68 BGP4 summary route information (Continued)

Field	Description
IBGP routes selected as best routes	The number of “best” routes in the BGP4 route table that are IBGP routes.
EBGP routes selected as best routes	The number of “best” routes in the BGP4 route table that are EBGP routes.

Displaying the BGP4 route table

BGP4 uses filters you define as well as the algorithm described in “How BGP4 selects a path for a route” on page 283 to determine the preferred route to a destination. BGP4 sends only the preferred route to the router IP table. However, if you want to view all the routes BGP4 knows about, you can display the BGP4 table using either of the following methods.

To view the BGP4 route table, enter the following command.

```
Brocade#show ip bgp routes
Total number of BGP Routes: 97371
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      10.8.1.0/24         192.168.4.106   0          100        0      BE
         AS_PATH: 65001
```

Syntax: **show ip bgp routes** [[**network**] *ip-addr*] | *num* | [**age** *secs*] | [**as-path-access-list** *num*] | [**best**] | [**cidr-only**] | [**community** *num* | **no-export** | **no-advertise** | **internet** | **local-as**] | [**community-access-list** *num*] | [**community-list** *num* | [**detail** *option*] | [**filter-list** *num, num,...*] | [**next-hop** *ip-addr*] | [**no-best**] | [**not-installed-best**] | [**prefix-list** *string*] | [**regular-expression** *regular-expression*] | [**route-map** *map-name*] | [**summary**] | [**unreachable**]

The *ip-addr* option displays routes for a specific network. The **network** keyword is optional. You can enter the network address without entering “network” in front of it.

The *num* option specifies the table entry with which you want the display to start. For example, if you want to list entries beginning with table entry 100, specify 100.

The **age** *secs* parameter displays only the routes that have been received or updated more recently than the number of seconds you specify.

The **as-path-access-list** *num* parameter filters the display using the specified AS-path ACL.

The **best** parameter displays the routes received from the neighbor that the Layer 3 switch selected as the best routes to their destinations.

The **cidr-only** option lists only the routes whose network masks do not match their class network length.

The **community** option lets you display routes for a specific community. You can specify **local-as**, **no-export**, **no-advertise**, **internet**, or a private community number. You can specify the community number as either two five-digit integer values of 1 through 65535, separated by a colon (for example, 12345:6789) or a single long integer value.

The **community-access-list** *num* parameter filters the display using the specified community ACL.

The **community-list** option lets you display routes that match a specific community filter.

The **detail** option lets you display more details about the routes. You can refine your request by also specifying one of the other display options after the detail keyword.

The **filter-list** option displays routes that match a specific address filter list.

The **next-hop** *ip-addr* option displays the routes for a given next-hop IP address.

The **no-best** option displays the routes for which none of the routes to a given prefix were selected as the best route. The IP route table does not contain a BGP4 route for any of the routes listed by the command.

The **not-installed-best** option displays the routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes).

The **prefix-list** *string* parameter filters the display using the specified IP prefix list.

The **regular-expression** *regular-expression* option filters the display based on a regular expression. Refer to "Using regular expressions to filter" on page 336.

The **route-map** *map-name* parameter filters the display using the specified route map. The software displays only the routes that match the match statements in the route map. The software disregards the route map set statements.

The **summary** option displays summary information for the routes.

The **unreachable** option displays the routes that are unreachable because the Layer 3 switch does not have a valid RIP, OSPF, or static route to the next hop.

Displaying the best BGP4 routes

To display all the BGP4 routes in the Layer 3 switch BGP4 route table that are the best routes to their destinations, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp routes best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      10.2.0.0/16         192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701
```

Syntax: show ip bgp routes best

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying the best BGP4 routes that are not in the IP route table

When the Layer 3 switch has multiple routes to a destination from different sources (such as BGP4, OSPF, RIP, or static routes), the Layer 3 switch selects the route with the lowest administrative distance as the best route, and installs that route in the IP route table.

To display the BGP4 routes that are the “best” routes to their destinations but are not installed in the Layer 3 switch IP route table, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp routes not-installed-best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      192.168.4.0/24     192.168.4.106   0          100        0      bE
         AS_PATH: 65001
```

Each of the displayed routes is a valid path to its destination, but the Layer 3 switch received another path from a different source (such as OSPF, RIP, or a static route) that has a lower administrative distance. The Layer 3 switch always selects the path with the lowest administrative distance to install in the IP route table.

Notice that the route status in this example is the new status, “b”. Refer to Table 69 on page 383 for a description.

Syntax: show ip bgp routes not-installed-best

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

NOTE
To display the routes that the Layer 3 switch has selected as the best routes and installed in the IP route table, display the IP route table using the **show ip route** command.

Displaying BGP4 routes whose destinations are unreachable

To display BGP4 routes whose destinations are unreachable using any of the BGP4 paths in the BGP4 route table, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp routes unreachable
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix      Next Hop        Metric     LocPrf     Weight Status
1      10.8.8.0/24  192.168.5.1     0          101        0
         AS_PATH: 65001 4355 1
```

Syntax: show ip bgp routes unreachable

For information about the fields in this display, refer to Table 69 on page 383. The fields in this display also appear in the **show ip bgp** display.

Displaying information for a specific route

To display BGP4 network information by specifying an IP address within the network, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp 10.3.4.0
Number of BGP Routes matching display condition : 1
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop        Metric LocPrf Weight Path
*>  10.3.4.0/24         192.168.4.106          100    0      65001 4355 1 1221 ?
       Last update to IP routing table: 0h11m38s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       1/2/1
       Route is advertised to 1 peers:
        10.20.20.2(65300)
```

Syntax: show ip bgp [**route**] *ip-addr/prefix* [**longer-prefixes**] | *ip-addr*

If you use the **route** option, the display for the information is different, as shown in the following example.

```
Brocade#show ip bgp route 10.3.4.0
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.3.4.0/24         192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1 1221
       Last update to IP routing table: 0h12m1s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       1/2/1
       Route is advertised to 1 peers:
        10.20.20.2(65300)
```

These displays show the following information.

TABLE 69 **BGP4 network information**

Field	Description
Number of BGP Routes matching display condition	The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command output. **NOTE:** This field appears only if you *do not* enter the **route** option.
Prefix	The network address and prefix.
Next Hop	The next-hop router for reaching the network from the Layer 3 switch.
Metric	The value of the route MED attribute. If the route does not have a metric, this field is blank.
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 through 4294967295.

TABLE 69 BGP4 network information (Continued)

Field	Description
Weight	The value that this router associates with routes from a specific neighbor. For example, if the router receives routes to the same destination from two BGP4 neighbors, the router prefers the route from the neighbor with the larger weight.
Path	The route AS path. **NOTE:** This field appears only if you *do not* enter the **route** option.
Origin code	A character the display uses to indicate the route origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command output. **NOTE:** This field appears only if you *do not* enter the **route** option.
Status	The route status, which can be one or more of the following: • A – AGGREGATE. The route is an aggregate route for multiple networks. • B – BEST. BGP4 has determined that this is the optimal route to the destination. **NOTE:** If the “b” is shown in lowercase, the software was not able to install the route in the IP route table. • b – NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes). • C – CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D – DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H – HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I – INTERNAL. The route was learned through BGP4. • L – LOCAL. The route originated on this Layer 3 switch. • M – MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with “B”. **NOTE:** If the “m” is shown in lowercase, the software was not able to install the route in the IP route table. • S – SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. **NOTE:** This field appears only if you enter the **route** option.

Displaying route details

Here is an example of the information displayed when you use the **detail** option. In this example, the information for one route is shown.

```
Brocade#show ip bgp routes detail
Total number of BGP Routes: 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
1       Prefix: 10.5.0.0/24,  Status: BME,  Age: 0h28m28s
         NEXT_HOP: 10.1.1.2,  Learned from Peer: 10.1.0.2 (5)
          LOCAL_PREF: 101,  MED: 0,  ORIGIN: igp,  Weight: 10
           AS_PATH: 5
            Adj_RIB_out count: 4,  Admin distance 20
```

These displays show the following information.

TABLE 70 BGP4 route information

Field	Description
Total number of BGP Routes	The number of BGP4 routes.
Status codes	A list of the characters the display uses to indicate the route status. The status code is appears in the left column of the display, to the left of each route. The status codes are described in the command output.
Prefix	The network prefix and mask length.
Status	The route status, which can be one or more of the following: • A – AGGREGATE. The route is an aggregate route for multiple networks. • B – BEST. BGP4 has determined that this is the optimal route to the destination. NOTE: If the “b” is shown in lowercase, the software was not able to install the route in the IP route table. • b – NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 switch received better routes from other sources (such as OSPF, RIP, or static IP routes). • C – CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D – DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H – HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I – INTERNAL. The route was learned through BGP4. • L – LOCAL. The route originated on this Layer 3 switch. • M – MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with “B”. NOTE: If the “m” is shown in lowercase, the software was not able to install the route in the IP route table. • S – SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.
Age	The last time an update occurred.
Next_Hop	The next-hop router for reaching the network from the Layer 3 switch.
Learned from Peer	The IP address of the neighbor that sent this route.
Local_Pref	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 through 4294967295.
MED	The route metric. If the route does not have a metric, this field is blank.
Origin	The source of the route information. The origin can be one of the following: • EGP – The routes with this set of attributes came to BGP through EGP. • IGP – The routes with this set of attributes came to BGP through IGP. • INCOMPLETE – The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.

TABLE 70 BGP4 route information (Continued)

Field	Description
Weight	The value that this router associates with routes from a specific neighbor. For example, if the router receives routes to the same destination from two BGP4 neighbors, the router prefers the route from the neighbor with the larger weight.
Atomic	Whether network information in this route has been aggregated *and* this aggregation has resulted in information loss. NOTE: Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Aggregation ID	The router that originated this aggregator.
Aggregation AS	The AS in which the network information was aggregated. This value applies only to aggregated routes and is otherwise 0.
Originator	The originator of the route in a route reflector environment.
Cluster List	The route-reflector clusters through which this route has passed.
Learned From	The IP address of the neighbor from which the Layer 3 switch learned the route.
Admin Distance	The administrative distance of the route.
Adj_RIB_out	The number of neighbors to which the route has been or will be advertised. This is the number of times the route has been selected as the best route and placed in the Adj-RIB-Out (outbound queue) for a BGP4 neighbor.
Communities	The communities the route is in.

Displaying BGP4 route-attribute entries

The route-attribute entries table lists the sets of BGP4 attributes stored in the router memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the router typically has fewer route attribute entries than routes. To display the route-attribute entries table, use one of the following methods.

To display the IP route table, enter the following command.

```
Brocade#show ip bgp attribute-entries
```

Syntax: **show ip bgp attribute-entries**

Here is an example of the information displayed by this command. A zero value indicates that the attribute is not set.

```
Brocade#show ip bgp attribute-entries
        Total number of BGP Attribute Entries: 7753
1       Next Hop  :192.168.11.1       Metric   :0              Origin:IGP
        Originator:0.0.0.0            Cluster List:None
         Aggregator:AS Number :0         Router-ID:0.0.0.0          Atomic:FALSE
        Local Pref:100                Communities:Internet
        AS Path   :(65002) 65001 4355 2548 3561 5400 6669 5548
2       Next Hop  :192.168.11.1       Metric   :0              Origin:IGP
        Originator:0.0.0.0            Cluster List:None
         Aggregator:AS Number :0         Router-ID:0.0.0.0          Atomic:FALSE
        Local Pref:100                Communities:Internet
        AS Path   :(65002) 65001 4355 2548
```

Table 71 lists the field definitions for the command output.

TABLE 71 BGP4 route-attribute entries information

Field	Description
Total number of BGP Attribute Entries	The number of routes contained in this router BGP4 route table.
Next Hop	The IP address of the next hop router for routes that have this set of attributes.
Metric	The cost of the routes that have this set of attributes.
Origin	The source of the route information. The origin can be one of the following: • EGP – The routes with this set of attributes came to BGP through EGP. • IGP – The routes with this set of attributes came to BGP through IGP. • INCOMPLETE – The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Originator	The originator of the route in a route reflector environment.
Cluster List	The route-reflector clusters through which this set of attributes has passed.
Aggregator	Aggregator information: • AS Number shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0. • Router-ID shows the router that originated this aggregator.
Atomic	Whether the network information in this set of attributes has been aggregated *and* this aggregation has resulted in information loss: • TRUE – Indicates information loss has occurred • FALSE – Indicates no information loss has occurred NOTE: Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Local Pref	The degree of preference for routes that use this set of attributes relative to other routes in the local AS.
Communities	The communities that routes with this set of attributes are in.
AS Path	The autonomous systems through which routes with this set of attributes have passed. The local AS is shown in parentheses.

Displaying the routes BGP4 has placed in the IP route table

The IP route table indicates the routes it has received from BGP4 by listing “BGP” as the route type.

To display the IP route table, enter the following command.

```
Brocade#show ip route
```

Syntax: show ip route [*ip-addr* | *num* | **bgp** | **ospf** | **rip**]

Here is an example of the information displayed by this command. Notice that most of the routes in this example have type “B”, indicating that their source is BGP4.

```
Brocade#show ip route
Total number of IP routes: 50834
B:BGP D:Directly-Connected  O:OSPF  R:RIP  S:Static
      Network Address  NetMask          Gateway          Port       Cost   Type
      10.0.0.0          255.0.0.0        192.168.13.2     1/1/1       0      B
      10.4.0.0          255.0.0.0        192.168.13.2     1/1/1       0      B
      10.20.0.0         255.255.128.0    192.168.13.2     1/1/1       0      B
      10.1.0.0         255.255.0.0      0.0.0.0          1/1/1      1      D
      10.10.11.0       255.255.255.0    0.0.0.0          1/1/4      1      D
      10.2.97.0        255.255.255.0    192.168.13.2     1/1/1      0      B
      10.3.63.0        255.255.255.0    192.168.13.2     1/1/1      0      B
      10.7.123.0       255.255.255.0    192.168.13.2     1/1/1      0      B
      10.5.252.0       255.255.254.0    192.168.13.2     1/1/1      0      B
      10.6.42.0        255.255.254.0    192.168.13.2     1/1/1      0      B
remaining 50824 entries not shown...
```

Displaying route flap dampening statistics

To display route dampening statistics or all the dampened routes, enter the following command at any level of the CLI.

```
Brocade#show ip bgp flap-statistics
Total number of flapping routes: 414
    Status Code  >:best d:damped h:history *:valid
    Network            From            Flaps Since    Reuse    Path
h>  192.168.206.0/23    10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  192.168.192.0/20   10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.168.165.0/24   10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.168.208.0/23    10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  192.168.0.0/16      10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  192.168.220.0/24    10.90.213.77    1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

Syntax: **show ip bgp flap-statistics** [**regular-expression** *regular-expression* | *address mask* [**longer-prefixes**] | **neighbor** *ip-addr* | **filter-list** *num*...]

The **regular-expression** *regular-expression* parameter is a regular expression. The regular expressions are the same ones supported for BGP4 AS-path filters. Refer to "Using regular expressions to filter" on page 336.

The *address mask* parameter specifies a particular route. If you also use the optional **longer-prefixes** parameter, then all statistics for routes that match the specified route or have a longer prefix than the specified route are displayed. For example, if you specify **10.157.0.0 longer**, then all routes with the prefix 10.157 or that have a longer prefix (such as 10.157.22) are displayed.

The **neighbor** *ip-addr* parameter displays route flap dampening statistics only for routes learned from the specified neighbor. You also can display route flap statistics for routes learned from a neighbor by entering the following command: **show ip bgp neighbors** *ip-addr* **flap-statistics.**

The **filter-list** *num* parameter specifies one or more filters. Only the routes that have been dampened and that match the specified filters are displayed.

Table 72 lists the field definitions for the command output.

TABLE 72 Route flap dampening statistics

Field	Description
Total number of flapping routes	The total number of routes in the Layer 3 switch BGP4 route table that have changed state and thus have been marked as flapping routes.
Status code	Indicates the dampening status of the route, which can be one of the following: • > – This is the best route among those in the BGP4 route table to the route destination. • d – This route is currently dampened, and thus unusable. • h – The route has a history of flapping and is unreachable now. • * – The route has a history of flapping but is currently usable.
Network	The destination network of the route.
From	The neighbor that sent the route to the Layer 3 switch.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time since the first flap of this route.
Reuse	The amount of time remaining until this route will be un-suppressed and thus be usable again.
Path	Shows the AS-path information for the route.

You also can display all the dampened routes by entering the following command.

show ip bgp dampened-paths.

Displaying the active route map configuration

To view the device active route map configuration (contained in the running-config) without displaying the entire running-config, enter the following command at any level of the CLI.

```
Brocade#show route-map
route-map permitnet4 permit 10
 match ip address prefix-list plist1
route-map permitnet1 permit 1
 match ip address prefix-list plist2
route-map setcomm permit 1
 set community 1234:2345 no-export
route-map test111 permit 111
 match address-filters 11
 set community 11:12 no-export
route-map permit1122 permit 12
 match ip address 11
route-map permit1122 permit 13
 match ip address std_22
```

This example shows that the running-config contains six route maps. Notice that the match and set statements within each route map are listed beneath the command for the route map itself. In this simplified example, each route map contains only one match or set statement.

To display the active configuration for a specific route map, enter a command such as the following, which specifies a route map name.

```
Brocade#show route-map setcomm
route-map setcomm permit 1
 set community 1234:2345 no-export
```

This example shows the active configuration for a route map called “setcomm“.

Syntax: **show route-map** [*map-name*]

Displaying BGP4 graceful restart neighbor information

Use the **show ip bgp neighbors** command to display BGP4 restart information for BGP4 neighbors.

```
Brocade# show ip bgp neighbors
    Total number of BGP Neighbors: 6
1   IP Address: 10.50.50.10, AS: 20 (EBGP), RouterID: 10.10.10.20
    State: ESTABLISHED, Time: 0h0m18s, KeepAliveTime: 60, HoldTime: 180
       KeepAliveTimer Expire in 34 seconds, HoldTimer Expire in 163 seconds
       Minimum Route Advertisement Interval: 0 seconds
       RefreshCapability: Received
       GracefulRestartCapability: Received
           Restart Time 120 sec, Restart bit 0
           afi/safi 1/1, Forwarding bit 0
       GracefulRestartCapability: Sent
           Restart Time 120 sec, Restart bit 0
           afi/safi 1/1, Forwarding bit 1
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
```

The text in bold is the BGP4 restart information for the specified neighbor.

Syntax: **show ip bgp neighbors**

Updating route information and resetting a neighbor session

The following sections describe ways to update route information with a neighbor, reset the session with a neighbor, and close a session with a neighbor.

Whenever you change a policy (ACL, route map, and so on) that affects the routes that the Layer 3 switch learns from a BGP4 neighbor or peer group of neighbors, you must enter a command to place the changes into effect. The changes take place automatically, but only affect new route updates. To make changes retroactive for routes received or sent before the changes were made, you need to enter a clear command.

You can update the learned routes using either of the following methods:

- Request the complete BGP4 route table from the neighbor or peer group. You can use this method if the neighbor supports the refresh capability (RFCs 2842 and 2858).
- Clear (reset) the session with the neighbor or peer group. This is the only method you can use if the neighbor does not support the refresh capability.

Each of these methods is effective, but can be disruptive to the network. The first method adds overhead while the Layer 3 switch learns and filters the neighbor or group entire route table, while the second method adds more overhead while the devices re-establish their BGP4 sessions.

You also can clear and reset the BGP4 routes that have been installed in the IP route table. Refer to “Clearing and resetting BGP4 routes in the IP route table” on page 397.

Using soft reconfiguration

The *soft reconfiguration* feature places policy changes into effect without resetting the BGP4 session. Soft reconfiguration does not request the neighbor or group to send its entire BGP4 table, nor does the feature reset the session with the neighbor or group. Instead, the soft reconfiguration feature stores all the route updates received from the neighbor or group. When you request a soft reset of inbound routes, the software performs route selection by comparing the policies against the stored route updates, instead of requesting the neighbor BGP4 route table or resetting the session with the neighbor.

When you enable the soft reconfiguration feature, it sends a refresh message to the neighbor or group if the neighbor or group supports dynamic refresh. Otherwise, the feature resets the neighbor session. This step is required to ensure that the soft reconfiguration feature has a complete set of updates to use, and occurs only once, when you enable the feature. The feature accumulates all the route updates from the neighbor, eliminating the need for additional refreshes or resets when you change policies in the future.

To use soft reconfiguration:

- Enable the feature.
- Make the policy changes.
- Apply the changes by requesting a soft reset of the inbound updates from the neighbor or group.

Use the following CLI methods to configure soft configuration, apply policy changes, and display information for the updates that are filtered out by the policies.

Enabling soft reconfiguration

To configure a neighbor for soft reconfiguration, enter a command such as the following.

```
Brocade(config-bgp-router)#neighbor 10.10.200.102 soft-reconfiguration inbound
```

This command enables soft reconfiguration for updates received from 10.10.200.102. The software dynamically refreshes or resets the session with the neighbor, then retains all route updates from the neighbor following the reset.

Syntax: [**no**] **neighbor** *ip-addr* | *peer-group-name* **soft-reconfiguration inbound**

NOTE
The syntax related to soft reconfiguration is shown. For complete command syntax, refer to "Adding BGP4 neighbors" on page 292.

Placing a policy change into effect

To place policy changes into effect, enter a command such as the following.

```
Brocade(config-bgp-router)#clear ip bgp neighbor 10.10.200.102 soft in
```

This command updates the routes by comparing the route policies against the route updates that the Layer 3 switch has stored. The command does not request additional updates from the neighbor or otherwise affect the session with the neighbor.

Syntax: **clear ip bgp neighbor** *ip-addr* | *peer-group-name* **soft in**

NOTE
If you do not specify "in", the command applies to both inbound and outbound updates.

NOTE
The syntax related to soft reconfiguration is shown. For complete command syntax, refer to "Dynamically refreshing routes" on page 394.

Displaying the filtered routes received from the neighbor or peer group

When you enable soft reconfiguration, the Layer 3 switch saves all updates received from the specified neighbor or peer group. This includes updates that contain routes that are filtered out by the BGP4 route policies in effect on the Layer 3 switch. To display the routes that have been filtered out, enter the following command at any level of the CLI.

```
Brocade#show ip bgp filtered-routes
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.0.0/8          192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 701 1 189
```

The routes displayed by the command are the routes that the Layer 3 switch BGP4 policies filtered out. The Layer 3 switch did not place the routes in the BGP4 route table, but did keep the updates. If a policy change causes these routes to be permitted, the Layer 3 switch does not need to request the route information from the neighbor, but instead uses the information in the updates.

Syntax: **show ip bgp filtered-routes** [*ip-addr*] | [**as-path-access-list** *num*] | [**detail**] | [**prefix-list** *string*]

The *ip-addr* parameter specifies the IP address of the destination network.

The **as-path-access-list** *num* parameter specifies an AS-path ACL. Only the routes permitted by the AS-path ACL are displayed.

The **detail** parameter displays detailed information for the routes. (The example above shows summary information.) You can specify any of the other options after **detail** to further refine the display request.

The **prefix-list** *string* parameter specifies an IP prefix list. Only the routes permitted by the prefix list are displayed.

NOTE
The syntax for displaying filtered routes is shown. For complete command syntax, refer to "Displaying the BGP4 route table" on page 380.

Displaying all the routes received from the neighbor

To display all the route information received in route updates from a neighbor since you enabled soft reconfiguration, enter a command such as the following at any level of the CLI.

```
Brocade#show ip bgp neighbors 192.168.4.106 received-routes
       There are 97345 received routes from  neighbor 192.168.4.106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106              100        0      BE
```

Syntax: **show ip bgp neighbors** *ip-addr* **received-routes** [**detail**]

The **detail** parameter displays detailed information for the routes. The example above shows summary information.

NOTE
The syntax for displaying received routes is shown. For complete command syntax, refer to "Displaying BGP4 neighbor information" on page 367.

NOTE
The **show ip bgp neighbors** *ip-addr* received-routes syntax supported in previous software releases is changed to the following syntax: **show ip bgp neighbors** *ip-addr* **routes.**

Dynamically requesting a route refresh from a BGP4 neighbor

You can easily apply changes to filters that control BGP4 routes received from or advertised to a neighbor, without resetting the BGP4 session between the Layer 3 switch and the neighbor. For example, if you add, change, or remove a BGP4 address filter that denies specific routes received from a neighbor, you can apply the filter change by requesting a route refresh from the neighbor. If the neighbor also supports dynamic route refreshes, the neighbor resends its Adj-RIB-Out, its table of BGP4 routes. Using the route refresh feature, you do not need to reset the session with the neighbor.

The route refresh feature is based on the following specifications:

- RFC 2842. This RFC specifies the Capability Advertisement, which a BGP4 router uses to dynamically negotiate a capability with a neighbor.
- RFC 2858 for Multi-protocol Extension.

  NOTE
  The Brocade implementation of dynamic route refresh supports negotiation of IP version 4 unicasts only.

- RFC 2918, which describes the dynamic route refresh capability

The dynamic route refresh capability is enabled by default and cannot be disabled. When the Layer 3 switch sends a BGP4 OPEN message to a neighbor, the Layer 3 switch includes a Capability Advertisement to inform the neighbor that the Layer 3 switch supports dynamic route refresh.

NOTE
The option for dynamically refreshing routes received from a neighbor requires the neighbor to support dynamic route refresh. If the neighbor does not support this feature, the option does not take effect and the software displays an error message. The option for dynamically re-advertising routes to a neighbor does not require the neighbor to support dynamic route refresh.

To use the dynamic refresh feature, use either of the following methods.

Dynamically refreshing routes

The following sections describe how to dynamically refresh BGP4 routes to place new or changed filters into effect.

To request a dynamic refresh of all routes from a neighbor, enter a command such as the following.

```
Brocade(config-bgp-router)#clear ip bgp neighbor 192.168.1.170 soft in
```

This command asks the neighbor to send its BGP4 table (Adj-RIB-Out) again. The Layer 3 switch applies its filters to the incoming routes and adds, modifies, or removes BGP4 routes as necessary.

Syntax: **clear ip bgp neighbor all** | *ip-addr* | *peer-group-name* | *as-num* [**soft-outbound** | **soft** [**in** | **out**]]

The **all** | *ip-addr* | *peer-group-name* | *as-num* option specifies the neighbor. The *ip-addr* parameter specifies a neighbor by its IP interface with the Layer 3 switch. The *peer-group-name* specifies all neighbors in a specific peer group. The *as-num* parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

The **soft-outbound** parameter updates all outbound routes by applying the new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

The **soft** [**in** | **out**] parameter specifies whether you want to refresh the routes received from the neighbor or sent to the neighbor:

- **soft in** does one of the following:
  - If you enabled soft reconfiguration for the neighbor or peer group, **soft in** updates the routes by comparing the route policies against the route updates that the Layer 3 switch has stored. Soft reconfiguration does not request additional updates from the neighbor or otherwise affect the session with the neighbor. Refer to "Using soft reconfiguration" on page 391.
  - If you did not enable soft reconfiguration, **soft in** requests the neighbor entire BGP4 route table (Adj-RIB-Out), then applies the filters to add, change, or exclude routes.
  - If a neighbor does not support dynamic refresh, **soft in** resets the neighbor session.
- **soft out** updates all outbound routes, then sends the Layer 3 switch entire BGP4 route table (Adj-RIB-Out) to the neighbor, after changing or excluding the routes affected by the filters.

If you do not specify **in** or **out**, the Layer 3 switch performs both options.

NOTE
The **soft-outbound** parameter updates all outbound routes by applying the new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor. The **soft out** parameter updates all outbound routes, then sends the Layer 3 switch entire BGP4 route table (Adj-RIB-Out) to the neighbor, after changing or excluding the routes affected by the filters. Use **soft-outbound** if only the outbound policy is changed.

To dynamically resend all the Layer 3 switch BGP4 routes to a neighbor, enter a command such as the following.

```
Brocade(config-bgp-router)#clear ip bgp neighbor 192.168.1.170 soft out
```

This command applies its filters for outgoing routes to the Layer 3 switch BGP4 route table (Adj-RIB-Out), changes or excludes routes accordingly, then sends the resulting Adj-RIB-Out to the neighbor.

NOTE
The Brocade Layer 3 switch does not automatically update outbound routes using a new or changed outbound policy or filter when a session with the neighbor goes up or down. Instead, the Layer 3 switch applies a new or changed policy or filter when a route is placed in the outbound queue (Adj-RIB-Out).

To place a new or changed outbound policy or filter into effect, you must enter a **clear ip bgp neighbor** command regardless of whether the neighbor session is up or down. You can enter the command without optional parameters or with the **soft out** or **soft-outbound** option. Either way, you must specify a parameter for the neighbor (*ip-addr*, *as-num*, peer-group-name, or **all**).

Displaying dynamic refresh information

You can use the **show ip bgp neighbors** command to display information for dynamic refresh requests. For each neighbor, the display lists the number of dynamic refresh requests the Layer 3 switch has sent to or received from the neighbor and indicates whether the Layer 3 switch received confirmation from the neighbor that the neighbor supports dynamic route refresh.

The RefreshCapability field indicates whether this Layer 3 switch has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability. The statistics in the Message Sent and Message Received rows under Refresh-Req indicate how many dynamic refreshes have been sent to and received from the neighbor. The statistic is cumulative across sessions.

```
Brocade#show ip bgp neighbors 10.4.0.2
1   IP Address: 10.4.0.2, AS: 5 (EBGP), RouterID: 10.10.10.1
       Description: neighbor 10.4.0.2
    State: ESTABLISHED, Time: 0h1m0s, KeepAliveTime: 0, HoldTime: 0
       PeerGroup: pg1
       Mutihop-EBGP: yes, ttl: 1
       RouteReflectorClient: yes
       SendCommunity: yes
       NextHopSelf: yes
       DefaultOriginate: yes (default sent)
       MaximumPrefixLimit: 90000
       RemovePrivateAs: : yes
       RefreshCapability: Received
    Route Filter Policies:
       Distribute-list: (out) 20
       Filter-list: (in) 30
       Prefix-list: (in) pf1
       Route-map: (in) setnp1  (out) setnp2
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       1       1         0            0
       Received: 1       8       1         0            0
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                  Tx: 0h0m59s    ---           Rx: 0h0m59s    ---
    Last Connection Reset Reason:Unknown
      Notification Sent:     Unspecified
      Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
       Byte Sent:   115, Received: 492
       Local host:  10.4.0.1, Local  Port: 179
       Remote host: 10.4.0.2, Remote Port: 8053
       ISentSeq:   52837276  SendNext:   52837392  TotUnAck:          0
       TotSent:         116  ReTrans:           0  UnAckSeq:   52837392
       IRcvSeq:  2155052043  RcvNext:  2155052536  SendWnd:       16384
       TotalRcv:        493  DupliRcv:          0  RcvWnd:        16384
       SendQue:           0  RcvQue:            0  CngstWnd:       1460
```

Closing or resetting a neighbor session

You can close a neighbor session or resend route updates to a neighbor.

If you make changes to filters or route maps and the neighbor does not support dynamic route refresh, use the following methods to ensure that neighbors contain only the routes you want them to contain:

- If you close a neighbor session, the Layer 3 switch and the neighbor clear all the routes they learned from each other. When the Layer 3 switch and neighbor establish a new BGP4 session, they exchange route tables again. Use this method if you want the Layer 3 switch to relearn routes from the neighbor and resend its own route table to the neighbor.
- If you use the soft-outbound option, the Layer 3 switch compiles a list of all the routes it would normally send to the neighbor at the beginning of a session. However, before sending the updates, the Brocade Layer 3 switch also applies the filters and route maps you have configured to the list of routes. If the filters or route maps result in changes to the list of routes, the Layer 3 switch sends updates to advertise, change, or even withdraw routes on the

neighbor as needed. This ensures that the neighbor receives only the routes you want it to contain. Even if the neighbor already contains a route learned from the Layer 3 switch that you later decided to filter out, using the soft-outbound option removes that route from the neighbor.

You can specify a single neighbor or a peer group.

To close a neighbor session and thus flush all the routes exchanged by the Layer 3 switch and the neighbor, enter the following command.

```
Brocade#clear ip bgp neighbor all
```

Syntax: **clear ip bgp neighbor all** | *ip-addr* | *peer-group-name* | *as-num* [**soft-outbound** | **soft** [**in** | **out**]]

The **all** | *ip-addr* | *peer-group-name* | *as-num* option specifies the neighbor. The *ip-addr* parameter specifies a neighbor by its IP interface with the Layer 3 switch. The *peer-group-name* specifies all neighbors in a specific peer group. The *as-num* parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

To resend routes to a neighbor without closing the neighbor session, enter a command such as the following.

```
Brocade#clear ip bgp neighbor 10.0.0.1 soft out
```

Clearing and resetting BGP4 routes in the IP route table

To clear BGP4 routes from the IP route table and reset the routes, enter a command such as the following.

```
Brocade#clear ip bgp routes
```

Syntax: **clear ip bgp routes** [*ip-addr/prefix-length*]

NOTE
The **clear ip bgp routes** command has the same effect as the **clear ip route** command, but applies only to routes that come from BGP4.

Clearing traffic counters

You can clear the counters (reset them to 0) for BGP4 messages. To do so, use one of the following methods.

To clear the BGP4 message counter for all neighbors, enter the following command.

```
Brocade#clear ip bgp traffic
```

Syntax: **clear ip bgp traffic**

To clear the BGP4 message counter for a specific neighbor, enter a command such as the following.

```
Brocade#clear ip bgp neighbor 10.0.0.1 traffic
```

To clear the BGP4 message counter for all neighbors within a peer group, enter a command such as the following.

```
Brocade#clear ip bgp neighbor PeerGroup1 traffic
```

Syntax: **clear ip bgp neighbor all** | *ip-addr* | *peer-group-name* | *as-num* **traffic**

The **all** | *ip-addr* | *peer-group-name* | *as-num* option specifies the neighbor. The *ip-addr* parameter specifies a neighbor by its IP interface with the Layer 3 switch. The *peer-group-name* specifies all neighbors in a specific peer group. The *as-num* parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

Clearing route flap dampening statistics

To clear route flap dampening statistics, use the following CLI method.

NOTE
Clearing the dampening statistics for a route does not change the dampening status of the route.

To clear all the route dampening statistics, enter the following command at any level of the CLI.

```
Brocade#clear ip bgp flap-statistics
```

Syntax: **clear ip bgp flap-statistics** [**regular-expression** *regular-expression* | *address mask* | **neighbor** *ip-addr*]

The parameters are the same as those for the **show ip bgp flap-statistics** command (except the **longer-prefixes** option is not supported). Refer to "Displaying route flap dampening statistics" on page 359.

NOTE
The **clear ip bgp damping** command not only clears statistics but also un-suppresses the routes. Refer to "Displaying route flap dampening statistics" on page 359.

Removing route flap dampening

You can un-suppress routes by removing route flap dampening from the routes. The Layer 3 switch allows you to un-suppress all routes at once or un-suppress individual routes.

To un-suppress all the suppressed routes, enter the following command at the Privileged EXEC level of the CLI.

```
Brocade#clear ip bgp damping
```

Syntax: **clear ip bgp damping** [*ip-addr ip-mask*]

The *ip-addr* parameter specifies a particular network.

The *ip-mask* parameter specifies the network mask.

To un-suppress a specific route, enter a command such as the following.

```
Brocade#clear ip bgp damping 10.157.22.0 255.255.255.0
```

This command un-suppresses only the routes for network 10.157.22.0/24.

Clearing diagnostic buffers

The Layer 3 switch stores the following BGP4 diagnostic information in buffers:

- The first 400 bytes of the last packet that contained an error
- The last NOTIFICATION message either sent or received by the Layer 3 switch

To display these buffers, use options with the **show ip bgp neighbors** command. Refer to "Displaying BGP4 neighbor information" on page 367.

This information can be useful if you are working with Brocade Technical Support to resolve a problem. The buffers do not identify the system time when the data was written to the buffer. If you want to ensure that diagnostic data in a buffer is recent, you can clear the buffers. You can clear the buffers for a specific neighbor or for all neighbors.

If you clear the buffer containing the first 400 bytes of the last packet that contained errors, all the bytes are changed to zeros. The Last Connection Reset Reason field of the BGP neighbor table also is cleared.

If you clear the buffer containing the last NOTIFICATION message sent or received, the buffer contains no data.

You can clear the buffers for all neighbors, for an individual neighbor, or for all the neighbors within a specific peer group.

To clear these buffers for neighbor 10.0.0.1, enter the following commands.

```
Brocade#clear ip bgp neighbor 10.0.0.1 last-packet-with-error
Brocade#clear ip bgp neighbor 10.0.0.1 notification-errors
```

Syntax: **clear ip bgp neighbor all** | *ip-addr* | *peer-group-name* | *as-num*
last-packet-with-error | **notification-errors**

The **all** | *ip-addr* | *peer-group-name* | *as-num* option specifies the neighbor. The *ip-addr* parameter specifies a neighbor by its IP interface with the Layer 3 switch. The *peer-group-name* specifies all neighbors in a specific peer group. The *as-num* parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

Chapter

IPv6

8

Table 73 lists the IPv6 features Brocade ICX 6650 devices support. These features are supported in the Layer 2 and full Layer 3 software images, except where explicitly noted.

TABLE 73 Supported IPv6 features

Feature	Brocade ICX 6650
IPv6 static routes	Yes
IPv6 over IPv4 tunnels	Yes
ECMP load sharing	Yes

Static IPv6 route configuration

You can configure a static IPv6 route to be redistributed into a routing protocol, but you cannot redistribute routes learned by a routing protocol into the static IPv6 routing table.

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface. For more information on performing these configuration tasks, refer to the section “Configuring IPv4 and IPv6 protocol stacks” in the *Brocade ICX 6650 Administration Guide*.

Configuring a static IPv6 route

To configure a static IPv6 route for a destination network with the prefix 2001:db8::0/32, a next-hop gateway with the global address 2001:db8:0:ee44::1, and an administrative distance of 110, enter the following command.

```
Brocade(config)#ipv6 route 2001:db8::0/32 2001:db8:0:ee44::1 distance 110
```

Syntax: **ipv6 route** *dest-ipv6-prefix/prefix-length next-hop-ipv6-address* [*metric*] [**distance** *number*]

To configure a static IPv6 route for a destination network with the prefix 2001:db8::0/32 and a next-hop gateway with the link-local address 2001:db8::1 that the Layer 3 switch can access through Ethernet interface 1/1/3, enter the following command.

```
Brocade(config)#ipv6 route 2001:db8::0/32 ethernet 1/1/3 2001:db8::1
```

Syntax: **ipv6 route** *dest-ipv6-prefix/prefix-length* [**ethernet** *stack-unit/slot/port* | **ve** *num*] *next-hop-ipv6-address* [*metric*] [**distance** *number*]

To configure a static IPv6 route for a destination network with the prefix 2001:db8::0/32 and a next-hop gateway that the Layer 3 switch can access through tunnel 1, enter the following command.

```
Brocade(config)#ipv6 route 2001:db8::0/32 tunnel 1
```

Syntax: **ipv6 route** *dest-ipv6-prefix/prefix-length interface port* [*metric*] [**distance** *number*]

Table 74 describes the parameters associated with this command and indicates the status of each parameter.

TABLE 74 Static IPv6 route parameters

Parameter	Configuration details	Status
The IPv6 prefix and prefix length of the route's destination network.	You must specify the *dest-ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.	Mandatory for all static IPv6 routes.
The route's next-hop gateway, which can be one of the following: • The IPv6 address of a next-hop gateway. • A tunnel interface.	You can specify the next-hop gateway as one of the following types of IPv6 addresses: • A global address. • A link-local address. If you specify a global address, you do not need to specify any additional parameters for the next-hop gateway. If you specify a link-local address, you must also specify the interface through which to access the address. You can specify one of the following interfaces: • An Ethernet interface. • A tunnel interface. • A virtual interface (VE). If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE or tunnel interface, also specify the VE or tunnel number. You can also specify the next-hop gateway as a tunnel interface. If you specify a tunnel interface, also specify the tunnel number.	Mandatory for all static IPv6 routes.
The route's metric.	You can specify a value from 1 – 16.	Optional for all static IPv6 routes. (The default metric is 1.)
The route's administrative distance.	You must specify the **distance** keyword and any numerical value.	Optional for all static IPv6 routes. (The default administrative distance is 1.)

A metric is a value that the Layer 3 switch uses when comparing this route to other static routes in the IPv6 static route table that have the same destination. The metric applies only to routes that the Layer 3 switch has already placed in the IPv6 static route table.

The administrative distance is a value that the Layer 3 switch uses to compare this route with routes from other route sources that have the same destination. (The Layer 3 switch performs this comparison before placing a route in the IPv6 route table.) This parameter does not apply to routes that are already in the IPv6 route table. In general, a low administrative distance indicates a preferred route. By default, static routes take precedence over routes learned by routing protocols. If you want a dynamic route to be chosen over a static route, you can configure the static route with a higher administrative distance than the dynamic route.

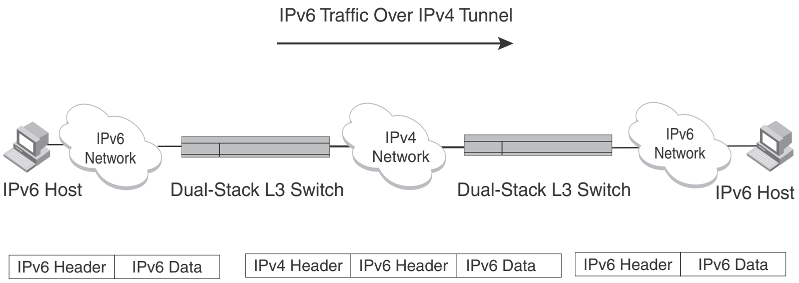
IPv6 over IPv4 tunnels

NOTE
This feature is supported only with the IPv6 Layer 3 PROM and the full Layer 3 image.

To enable communication between isolated IPv6 domains using the IPv4 infrastructure, you can manually configure IPv6 over IPv4 tunnels that provide static point-point connectivity.

As shown in Figure 29, these tunnels encapsulate an IPv6 packet within an IPv4 packet.

FIGURE 29 IPv6 over an IPv4 tunnel



In general, a manually configured tunnel establishes a permanent link between switches in IPv6 domains. A manually configured tunnel has explicitly configured IPv4 addresses for the tunnel source and destination.

This tunneling mechanism requires that the Layer 3 switch at each end of the tunnel run both IPv4 and IPv6 protocol stacks. The Layer 3 switches running both protocol stacks, or dual-stack routers, can interoperate directly with both IPv4 and IPv6 end systems and routers. Refer to to the section “Configuring IPv4 and IPv6 protocol stacks” in the *Brocade ICX 6650 Administration Guide*.

IPv6 over IPv4 tunnel configuration notes

- The local tunnel configuration must include both source and destination addresses.
- The remote side of the tunnel must have the opposite source/destination pair.
- A tunnel interface supports static and dynamic IPv6 configuration settings and routing protocols.
- Duplicate Address Detection (DAD) is not currently supported with IPv6 tunnels. Make sure tunnel endpoints do not have duplicate IP addresses.
- Neighbor Discovery (ND) is not supported with IPv6 tunnels.
- If a tunnel source port is a multi-homed IPv4 source, the tunnel will use the first IPv4 address only. For proper tunnel operation, use the ip address option.

Configuring a manual IPv6 tunnel

You can use a manually configured tunnel to connect two isolated IPv6 domains. You should deploy this point-to-point tunnelling mechanism if you need a permanent and stable connection.

To configure a manual IPv6 tunnel, enter commands such as the following on a Layer 3 Switch running both IPv4 and IPv6 protocol stacks on each end of the tunnel.

```
Brocade(config)#interface tunnel 1
Brocade(config-tnif-1)#tunnel source ethernet 1/1/1
Brocade(config-tnif-1)#tunnel destination 198.168.100.1
Brocade(config-tnif-1)#tunnel mode ipv6ip
Brocade(config-tnif-1)#ipv6 enable
```

This example creates tunnel interface 1 and assigns a link local IPv6 address with an automatically computed EUI-64 interface ID to it. The IPv4 address assigned to Ethernet interface 1/1/1 is used as the tunnel source, while the IPv4 address 192.168.100.1 is configured as the tunnel destination. The tunnel mode is specified as a manual IPv6 tunnel. Finally, the tunnel is enabled. Note that instead of entering **ipv6 enable**, you could specify an IPv6 address, for example, **ipv6 address 2001:b78:384d:34::/64 eui-64**, which would also enable the tunnel.

Syntax: **[no] interface tunnel** *number*

For the *number* parameter, specify a value between 1–8.

Syntax: **[no] tunnel source** *ipv4-address* **| ethernet** *port* **| loopback** *number* **| ve** *number*

The tunnel source can be an IP address or an interface.

For *ipv4-address*, use 8-bit values in dotted decimal notation.

The **ethernet | loopback | ve** parameter specifies an interface as the tunnel source. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a loopback, VE, or interface, also specify the loopback, VE, or number, respectively.

Syntax: **[no] tunnel destination** *ipv4-address*

Specify the *ipv4-address* parameter using 8-bit values in dotted decimal notation.

Syntax: **[no] tunnel mode ipv6ip**

ipv6ip indicates that this is an IPv6 manual tunnel.

Syntax: **ipv6 enable**

The **ipv6 enable** command enables the tunnel. Alternatively, you could specify an IPv6 address, which would also enable the tunnel.

Syntax: **ipv6 address** *ipv6-prefix/prefix-length* **[eui-64]**

The **ipv6 address** command enables the tunnel. Alternatively, you could enter **ipv6 enable**, which would also enable the tunnel.

Specify the *ipv6-prefix* parameter in hexadecimal format using 16-bit values between colons as documented in RFC 2373.

Specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter. The **eui-64** keyword configures the global address with an EUI-64 interface ID in the low-order 64 bits. The interface ID is automatically constructed in IEEE EUI-64 format using the interface's MAC address.

Clearing IPv6 tunnel statistics

You can clear statistics (reset all fields to zero) for all IPv6 tunnels or for a specific tunnel interface.

For example, to clear statistics for tunnel 1, enter the following command at the Privileged EXEC level or any of the Config levels of the CLI.

```
Brocade#clear ipv6 tunnel 1
```

To clear statistics for all IPv6 tunnels, enter the following command.

```
Brocade#clear ipv6 tunnel
```

Syntax: **clear ipv6 tunnel** [*number*]

The *number* parameter specifies the tunnel number.

Displaying IPv6 tunnel information

Use the commands in this section to display the configuration, status, and counters associated with IPv6 tunnels.

Displaying a summary of tunnel information

To display a summary of tunnel information, enter the following command at any level of the CLI.

```
Brocade#show ipv6 tunnel
IP6 Tunnels
  Tunnel  Mode         Packet Received  Packet Sent
  1       configured   0                0
  2       configured   0                22419
```

Syntax: **show ipv6 tunnel**

This display shows the following information.

TABLE 75 IPv6 tunnel summary information

Field	Description
Tunnel	The tunnel interface number.
Mode	The tunnel mode. Possible modes include the following: • configured – Indicates a manually configured tunnel.
Packet Received	The number of packets received by a tunnel interface. Note that this is the number of packets received by the CPU. It does not include the number of packets processed in hardware.
Packet Sent	The number of packets sent by a tunnel interface. Note that this is the number of packets sent by the CPU. It does not include the number of packets processed in hardware.

Displaying tunnel interface information

To display status and configuration information for tunnel interface 1, enter the following command at any level of the CLI.

```
Brocade#show interfaces tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Tunnel source ve 30
  Tunnel destination is 10.2.2.10
  Tunnel mode ipv6ip
  No port name
  MTU 1480 bytes, encapsulation IPV4
```

Syntax: **show interfaces tunnel** *number*

The *number* parameter indicates the tunnel interface number for which you want to display information.

This display shows the following information.

TABLE 76 IPv6 tunnel interface information

Field	Description
Tunnel interface status	The status of the tunnel interface can be one of the following: • **up** – The tunnel mode is set and the tunnel interface is enabled. • **down** – The tunnel mode is not set. • **administratively down** – The tunnel interface was disabled with the **disable** command.
Line protocol status	The status of the line protocol can be one of the following: • **up** – IPv4 connectivity is established. • **down** – The line protocol is not functioning and is down.
Hardware is tunnel	The interface is a tunnel interface.
Tunnel source	The tunnel source can be one of the following: • An IPv4 address • The IPv4 address associated with an interface/port.
Tunnel destination	The tunnel destination can be an IPv4 address.
Tunnel mode	The tunnel mode can be the following: • **ipv6ip** – indicates a manually configured tunnel
Port name	The port name configured for the tunnel interface.
MTU	The setting of the IPv6 maximum transmission unit (MTU).

Displaying interface level IPv6 settings

To display Interface level IPv6 settings for tunnel interface 1, enter the following command at any level of the CLI.

```
Brocade#show ipv6 inter tunnel 1
Interface Tunnel 1 is up, line protocol is up
  IPv6 is enabled, link-local address is 2001:db8::3:4:2 [Preferred]
  Global unicast address(es):
2001:db8::1 [Preferred],  subnet is 2001:db8::/64
2001:db8::1 [Preferred],  subnet is 2001:db8::/64
  Joined group address(es):
2001:db8::1:ff04:2
2001:db8::5
2001:db8::1:ff00:1
2001:db8::2
2001:db8::1
  MTU is 1480 bytes
  ICMP redirects are enabled
  No Inbound Access List Set
  No Outbound Access List Set
  OSPF enabled
```

The display command above reflects the following configuration.

```
Brocade#show running-config interface tunnel 1
!
interface tunnel 1
 port-name ManualTunnel1
 tunnel mode ipv6ip
 tunnel source loopback 1
 tunnel destination 10.1.1.1
 ipv6 address 2001:db8::1/64
 ipv6 address 2001:db8::1/64
 ipv6 ospf area 0
```

This display shows the following information.

TABLE 77 Interface level IPv6 tunnel information

Field	Description
Interface Tunnel status	The status of the tunnel interface can be one of the following: • **up** – IPv4 connectivity is established. • **down** – The tunnel mode is not set. • **administratively down** – The tunnel interface was disabled with the **disable** command.
Line protocol status	The status of the line protocol can be one of the following: • **up** – IPv6 is enabled through the **ipv6 enable** or **ipv6 address** command. • **down** – The line protocol is not functioning and is down.

ECMP load sharing for IPv6

The IPv6 route table selects the best route to a given destination from among the routes in the tables maintained by the configured routing protocols (BGP4, OSPF, static, and so on). The IPv6 route table can contain more than one path to a given destination. When this occurs, the Brocade device selects the path with the lowest cost for insertion into the routing table. If more than one path with the lowest cost exists, all of these paths are inserted into the routing table, subject to the configured maximum number of load sharing paths (by default 4). The device uses ***Equal-Cost Multi-Path (ECMP) load sharing*** to select a path to a destination.

When a route is installed by routing protocols or configured static route for the first time, and the IPv6 route table contains multiple, equal-cost paths to that route, the device checks the IPv6 neighbor for each next hop. Every next hop where the link layer address has been resolved will be stored in hardware. The device will initiate neighbor discovery for the next hops whose link layer addresses are not resolved. The hardware will hash the packet and choose one of the paths. The number of paths would be updated in hardware as the link layer gets resolved for a next hop.

If the path selected by the device becomes unavailable, the IPv6 neighbor should change state and trigger the update of the destination in the hardware.

Brocade devices support network-based ECMP load-sharing methods for IPv6 traffic. The Brocade device distributes traffic across equal-cost paths based on a XOR of some bits from the MAC source address, MAC destination address, IPv6 source address, IPv6 destination address, IPv6 flow label, IPv6 next header. The software selects a path based on a calculation involving the maximum number of load-sharing paths allowed and the actual number of paths to the destination network. This is the default ECMP load-sharing method for IPv6.

You can manually disable or enable ECMP load sharing for IPv6 and specify the number of equal-cost paths the device can distribute traffic across. In addition, you can display information about the status of ECMP load-sharing on the device.

Disabling or re-enabling ECMP load sharing for IPv6

ECMP load sharing for IPv6 is enabled by default. To disable the feature, enter the following command.

```
Brocade(config)#no ipv6 load-sharing
```

If you want to re-enable the feature after disabling it, you must specify the number of load-sharing paths. The maximum number of paths the device supports is a value from 2–8. By entering a command such as the following, iPv6 load-sharing will be re-enabled.

```
Brocade(config)#ipv6 load-sharing 4
```

Syntax: [no] ipv6 load-sharing *num*

The *num* parameter specifies the number of paths and can be from 2-8. The default is 4.

Changing the maximum load sharing paths for IPv6

By default, IPv6 ECMP load sharing allows traffic to be balanced across up to four equal paths. You can change the maximum number of paths the device supports to a value from 2-8.

To change the number of ECMP load sharing paths for IPv6, enter a command such as the following.

```
Brocade(config)#ipv6 load-sharing 6
```

Syntax: [no] ipv6 load-sharing [*num*]

The *num* parameter specifies the number of paths and can be from 2-8. The default is 4.

Enabling support for network-based ECMP load sharing for IPv6

Network-based ECMP load sharing is supported. In this configuration, traffic is distributed across equal-cost paths based on the destination network address. Routes to each network are stored in CAM and accessed when a path to a network is required. Because multiple hosts are likely to reside on a network, this method uses fewer CAM entries.

Displaying ECMP load-sharing information for IPv6

To display the status of ECMP load sharing for IPv6, enter the following command.

```
Brocade#show ipv6
Global Settings
  unicast-routing enabled, hop-limit 64
  No Inbound Access List Set
  No Outbound Access List Set
  Prefix-based IPv6 Load-sharing is Enabled, Number of load share paths: 4
```

Chapter

VRRP and VRRP-E 9

Table 78 lists the Virtual Router Redundancy Protocol (VRRP) and Virtual Router Redundancy Protocol Extended (VRRP-E) features Brocade ICX 6650 devices support.

NOTE
VRRP and VRRP-E is supported Brocade ICX 6650 devices that are running the full Layer 3 image.

TABLE 78 Supported VRRP and VRRP-E features

Feature	Brocade ICX 6650
Virtual Router Redundancy Protocol (VRRP)	Yes
VRRP timer scaling	Yes
VRRP Extended (VRRP-E)	Yes
IPv6 VRRP-E	Yes
IPv6 VRRP v3	Yes
VRRP-E slow start timer	Yes
VRRP-E timer scale	Yes
Forcing a Master router to abdicate to a standby router	Yes

* Refers to support for only IPv6 modules for these devices.

This chapter describes how to configure Brocade Layer 3 switches with the following router redundancy protocols:

- Virtual Router Redundancy Protocol (VRRP) – The standard router redundancy protocol described in RFC 2338. The Brocade ICX 6650 devices support VRRP version 2 (v2) and VRRP version 3 (v3). VRRP v2 supports the IPv4 environment, and VRRP v3 supports the IPv6 environment.
- VRRP Extended (VRRP-E) – An enhanced version of VRRP that overcomes limitations in the standard protocol. The Brocade ICX 6650 devices support VRRP-E v2 and VRRP-E v3. VRRP-E v2 supports the IPv4 environment, and VRRP-E v3 supports the IPv6 environment.

NOTE
VRRP and VRRP-E are separate protocols. You cannot use them together.

NOTE
You can use a Brocade Layer 3 switch configured for VRRP with another Brocade Layer 3 switch or a third-party router that is also configured for VRRP. However, you can use only a Brocade Layer 3 switch configured for VRRP-E only with another Brocade Layer 3 switch that also is configured for VRRP-E.

NOTE
The maximum number of supported VRRP or VRRP-E router instances is 254 for IPv4 environments. The maximum number of supported VRRP or VRRP-E router instances is 128 for IPv6 environments.

For a summary of how these two router redundancy protocols differ, refer to "Comparison of VRRP and VRRP-E" on page 420.

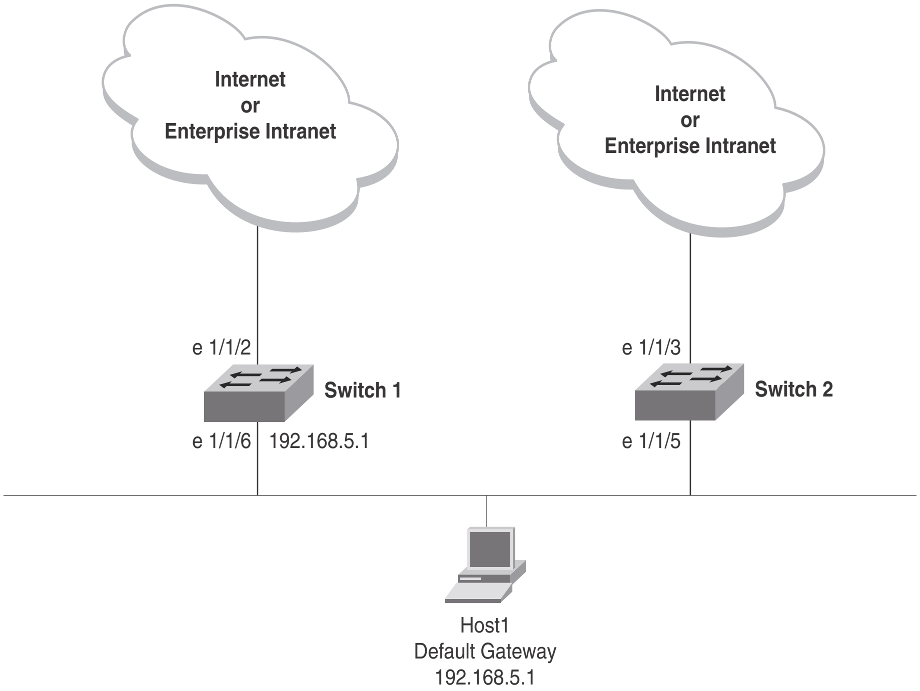
VRRP and VRRP-E overview

The following sections describe VRRP and VRRP-E. The protocols both provide redundant paths for IP addresses. However, the protocols differ in a few important ways. For clarity, each protocol is described separately.

VRRP overview

Virtual Router Redundancy Protocol (VRRP) provides redundancy to routers within a LAN. VRRP allows you to provide alternate router paths for a host without changing the IP address or MAC address by which the host knows its gateway. Consider the situation shown in Figure 30.

FIGURE 30 Switch 1 is the Host1 default gateway but is a single point of failure

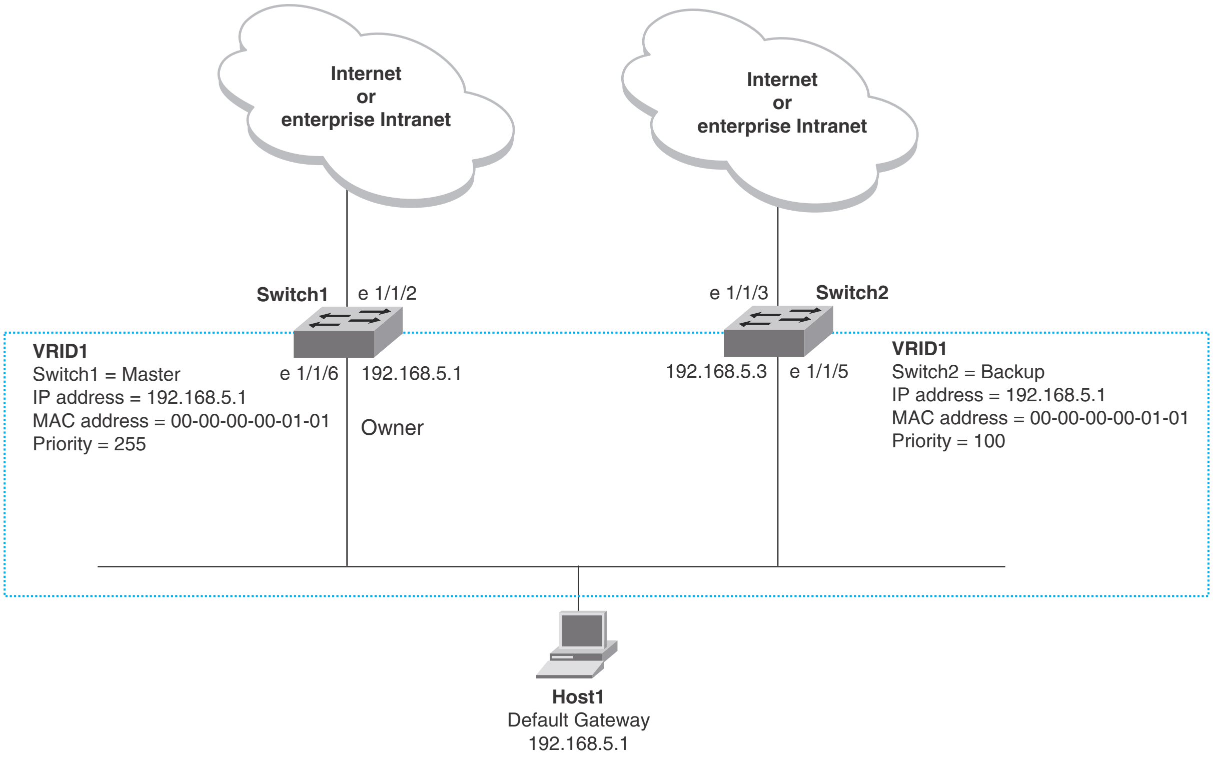


Switch 1 is the host default gateway out of the subnet. If this interface goes down, Host1 is cut off from the rest of the network. Switch 1 is thus a single point of failure for Host1's access to other networks.

If Switch 1 fails, you could configure Host1 to use Switch 2. Configuring one host with a different default gateway might not require too much extra administration. However, consider a more realistic network with dozens or even hundreds of hosts per subnet; reconfiguring the default gateways for all the hosts is impractical. It is much simpler to configure a VRRP virtual router on Switch 1 and Switch 2 to provide a redundant path for the hosts.

Figure 31 shows the same example network shown in Figure 30, but with a VRRP virtual router configured on Switch 1 and Switch 2.

FIGURE 31 Switch 1 and Switch 2 configured as VRRP virtual routers for redundant network access for Host1



The dashed box in Figure 31 represents a VRRP virtual router. When you configure a virtual router, one of the configuration parameters is the virtual router ID (VRID), which can be a number from 1 through 255. In this example, the VRID is 1.

NOTE
You can provide more redundancy by also configuring a second VRID with Switch 2 as the Owner and Switch 1 as the Backup. This type of configuration is sometimes called *Multigroup VRRP.*

Virtual router ID

A virtual router ID (VRID) consists of one Master router and one or more Backup routers. The Master router is the router that owns the IP addresses you associate with the VRID. For this reason, the Master router is sometimes called the “Owner”. Configure the VRID on the router that owns the default gateway interface. The other router in the VRID does not own the IP addresses associated with the VRID but provides the backup path if the Master router becomes unavailable.

Virtual router MAC address

Notice the MAC address associated with VRID1 in Figure 31. The first five octets of the address are the standard MAC prefix for VRRP packets, as described in RFC 2338. The last octet is the VRID. The VRID number becomes the final octet in the virtual MAC address associated with the virtual router.

When you configure a VRID, the software automatically assigns its MAC address. When a VRID becomes active, the Master router broadcasts a gratuitous ARP request containing the virtual router MAC address for each IP address associated with the virtual router. In Figure 31, Switch 1 sends a gratuitous ARP request with MAC address 00-00-5E-00-01-01 and IP address 192.168.5.1. Hosts use the virtual router MAC address in routed traffic they send to their default IP gateway (in this example, 192.168.5.1).

Virtual router IP address

VRRP does not use virtual IP addresses. Thus, there is no virtual IP address associated with a virtual router. Instead, you associate the virtual router with one or more real interface IP addresses configured on the router that owns the real IP addresses. In Figure 31, the virtual router with VRID1 is associated with real IP address 192.168.5.1, which is configured on interface e1/1/6 on Switch 1. VRIDs are interface-level parameters, not system-level parameters, so the IP address you associate with the VRID must already be a real IP address configured on the Owner interface.

NOTE
You can associate a virtual router with a virtual interface. A virtual interface is a named set of physical interfaces.

When you configure the Backup router for the VRID, specify the same IP address as the one you specify on the Owner. This is the IP address used by the host as its default gateway. The IP address cannot also exist on the Backup router. The interface on which you configure the VRID on the Backup router must have an IP address in the same subnet.

NOTE
If you delete a real IP address used by a VRRP entry, the VRRP entry also is deleted automatically.

NOTE
When a Backup router takes over forwarding responsibilities from a failed Master router, the Backup forwards traffic addressed to the VRID MAC address, which the host believes is the MAC address of the router interface for its default gateway. However, the Backup router cannot reply to IP pings sent to the IP addresses associated with the VRID. Because the IP addresses are owned by the Owner, if the Owner is unavailable, the IP addresses are unavailable as packet destinations.

Master negotiation

The routers within a VRID use the VRRP priority values associated with each router to determine which router becomes the Master. When you configure the VRID on a router interface, you specify whether the router is the Owner of the IP addresses you plan to associate with the VRID or a Backup router. If you indicate that the router is the Owner of the IP addresses, the software automatically sets the router VRRP priority for the VRID to 255, the highest VRRP priority. The router with the highest priority becomes the Master.

Backup routers can have a priority from 3 through 254, which you assign when you configure the VRID on the Backup router interfaces. The default VRRP priority for Backup routers is 100.

Because the router that owns the IP addresses associated with the VRID always has the highest priority, when all the routers in the virtual router are operating normally, the negotiation process results in the Owner of the VRID IP addresses becoming the Master router. Thus, the VRRP negotiation results in the normal case, in which the host's path to the default route is to the router that owns the interface for that route.

Hello messages

Virtual routers use Hello messages for negotiation to determine the Master router. Virtual routers send Hello messages to IP Multicast address 224.0.0.18. The frequency with which the Master sends Hello messages is the Hello interval. Only the Master sends Hello messages. However, a Backup router uses the Hello interval you configure for the Backup router if it becomes the Master.

The Backup routers wait for a period of time called the dead interval for a Hello message from the Master. If a Backup router does not receive a Hello message by the time the dead interval expires, the Backup router assumes that the Master router is dead and negotiates with the other Backup routers to select a new Master router. The Backup router with the highest priority becomes the new Master.

Master and Owner backup routers

If the Owner becomes unavailable, but then comes back online, the Owner again becomes the Master router. The Owner becomes the Master router again because it has the highest priority. The Owner always becomes the Master again when the Owner comes back online.

NOTE
If you configure a track port on the Owner and the track port is down, the Owner priority is changed to the track priority. In this case, the Owner does not have a higher priority than the Backup router that is acting as the Master router and the Owner therefore does not resume its position as the Master router. For more information about track ports, refer to "Track ports and track priority" on page 416.

By default, if a Backup is acting as the Master, and the original Master is still unavailable, another Backup can "preempt" the Backup that is acting as the Master. This can occur if the new Backup router has a higher priority than the Backup router that is acting as the Master. You can disable this behavior. When you disable preemption, a Backup router that has a higher priority than the router that is currently acting as the Master does not preempt the new Master by initiating a new Master negotiation. Refer to "Backup preempt configuration" on page 440.

NOTE
Regardless of the setting for the preempt parameter, the Owner always becomes the Master again when it comes back online.

Track ports and track priority

The Brocade implementation of VRRP enhances the protocol by giving a VRRP router the capability to monitor the state of the interfaces on the other end of the route path through the router. For example, in Figure 31 on page 413, interface e1/1/6 on Switch 1 owns the IP address to which Host1 directs route traffic on its default gateway. The exit path for this traffic is through the Switch 1 e1/1/2 interface.

Suppose interface e1/1/2 goes down. Even if interface e1/1/6 is still up, Host1 is cut off from other networks. In conventional VRRP, Switch 1 would continue to be the Master router despite the unavailability of the exit interface for the path the router is supporting. However, if you configure interface e1/1/6 to track the state of interface e1/1/2, if e1/1/2 goes down, interface e1/1/6 responds by changing the Switch 1 VRRP priority to the value of the track priority. In the configuration shown in Figure 31 on page 413, the Switch 1 priority changes from 255 to 20. One of the parameters contained in the Hello messages the Master router sends to its Backup routers is the Master router priority. If the track port feature results in a change in the Master router priority, the Backup routers quickly become aware of the change and initiate a negotiation to become the Master router.

In Figure 31 on page 413, the track priority results in the Switch 1 VRRP priority becoming lower than the Switch 2 VRRP priority. As a result, when Switch 2 learns that it now has a higher priority than Switch 1, Switch 2 initiates negotiation to become the Master router and becomes the new Master router, thus providing an open path for the Host1 traffic. To take advantage of the track port feature, make sure the track priorities are always lower than the VRRP priorities. The default track priority for the router that owns the VRID IP addresses is 2. The default track priority for Backup routers is 1. If you change the track port priorities, make sure you assign a higher track priority to the Owner of the IP addresses than the track priority you assign on the Backup routers.

Suppression of RIP advertisements for backed-up interfaces

The Brocade implementation also enhances VRRP by allowing you to configure the protocol to suppress RIP advertisements for the backed-up paths from Backup routers. Normally, a VRRP Backup router includes route information for the interface it is backing up in RIP advertisements. As a result, other routers receive multiple paths for the interface and might sometimes unsuccessfully use the path to the Backup router rather than the path to the Master router. If you enable the Brocade implementation of VRRP to suppress the VRRP Backup routers from advertising the backed-up interface in RIP, other routers learn only the path to the Master router for the backed-up interface.

Authentication

The Brocade implementations of VRRP and VRRP-E can use simple passwords to authenticate VRRP and VRRP-E packets. VRRP-E can also use HMAC-MD5-96 to authenticate VRRP-E packets.

VRRP and VRRP-E authentication is configured on the router interfaces. The VRRP authentication configuration of every router interface must match. For example, if you want to use simple passwords to authenticate VRRP traffic within a router, you must configure VRRP simple password authentication with the same password on all of the participating router interfaces.

NOTE
The HMAC-MD5-96 authentication type is supported for VRRP-E, but not supported for VRRP.

Independent operation of VRRP alongside RIP, OSPF, and BGP4

VRRP operation is independent of RIP, OSPF, and BGP4; therefore, RIP, OSPF, and BGP4 are not affected if VRRP is enabled on one of these interfaces.

Dynamic VRRP configuration

All VRRP global and interface parameters take effect immediately. You do not need to reset the system to place VRRP configuration parameters into effect.

VRRP-E overview

The most important difference between VRRP and VRRP-E is that all VRRP-E routers are Backup routers; there is no Owner router. VRRP-E overcomes the limitations in standard VRRP by removing the Owner.

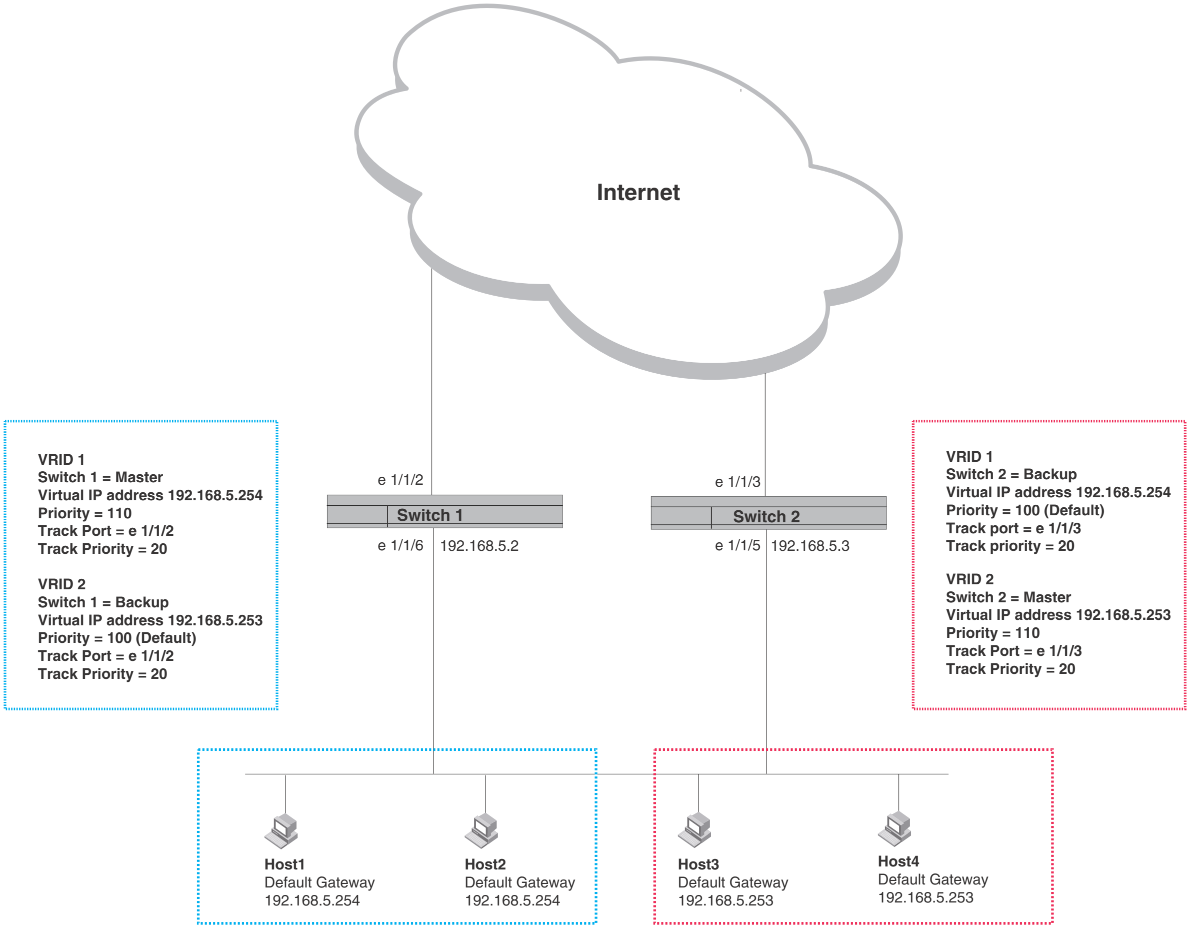
The following points explain how VRRP-E differs from VRRP:

- Owners and Backup routers
    - VRRP has an Owner and one or more Backup routers for each VRID. The Owner is the router on which the VRID's IP address is also configured as a real address. All the other routers supporting the VRID are Backup routers.
    - VRRP-E does not use Owners. All routers are Backup routers for a given VRID. The router with the highest priority becomes the Master. If there is a tie for highest priority, the router with the highest IP address becomes the Master. The elected Master owns the virtual IP address and answers pings and ARP requests.
- VRID's IP address
    - VRRP requires that the VRID’s IP address also be a real IP address configured on the VRID's interface on the Owner.
    - VRRP-E requires only that the VRID be in the same subnet as an interface configured on the VRID's interface. VRRP-E does not allow you to specify a real IP address configured on the interface as the VRID IP address.
- VRID's MAC address
    - VRRP uses the source MAC address as a virtual MAC address defined as 00-00-5E-00-01-*vrid*, where *vrid* is the VRID. The Master owns the virtual MAC address.
    - VRRP-E uses the MAC address of the interface as the source MAC address. The MAC address is 02-E0-52-*hash-value-vrid*, where *hash-value* is a two-octet hashed value for the IP address and *vrid* is the VRID.

- Hello packets
  - VRRP sends Hello messages to IP Multicast address 224.0.0.18.
  - VRRP-E uses UDP to send Hello messages in IP multicast messages. The Hello packets use the MAC address of the interface and the IP address as the source addresses. The destination MAC address is 00-00-00-00-00-02, and the destination IP address is 224.0.0.2 (the well-known IP multicast address for “all routers”). Both the source and destination UDP port number is 8888. VRRP-E messages are encapsulated in the data portion of the packet.
- Track ports and track priority
  - VRRP changes the priority of the VRID to the track priority, which typically is lower than the VRID priority and lower than the VRID priorities configured on the Backup routers. For example, if the VRRP interface priority is 100 and a tracked interface with track priority 20 goes down, the software changes the VRRP interface priority to 20.
  - VRRP-E reduces the priority of a VRRP-E interface by the amount of a tracked interface priority if the tracked interface link goes down. For example, if the VRRP-E interface priority is 200 and a tracked interface with track priority 20 goes down, the software changes the VRRP-E interface priority to 180. If another tracked interface goes down, the software reduces the VRID priority again, by the amount of the tracked interface track priority.
- VRRP-E can use HMAC-MD5-96 for authenticating VRRP-E packets. VRRP can use only simple passwords.

Figure 32 shows an example of a VRRP-E configuration.

FIGURE 32 Switch 1 and Switch 2 are configured to provide dual redundant network access for the host



In this example, Switch 1 and Switch 2 use VRRP-E to load share as well as provide redundancy to the hosts. The load sharing is accomplished by creating two VRRP-E groups. Each group has its own virtual IP addresses. Half of the clients point to VRID 1's virtual IP address as their default gateway and the other half point to VRID 2's virtual IP address as their default gateway. This organization enables some of the outbound Internet traffic to go through Switch 1 and the rest to go through Switch 2.

Switch 1 is the Master router for VRID 1 (backup priority = 110) and Switch 2 is the Backup router for VRID 1 (backup priority = 100). Switch 1 and Switch 2 both track the uplinks to the Internet. If an uplink failure occurs on Switch 1, its backup priority is decremented by 20 (track priority = 20), so that all traffic destined to the Internet is sent through Switch 2 instead.

Similarly, Switch 2 is the Master router for VRID 2 (backup priority = 110) and Switch 1 is the Backup router for VRID 2 (backup priority = 100). Switch 1 and Switch 2 are both tracking the uplinks to the Internet. If an uplink failure occurs on Switch 2, its backup priority is decremented by 20 (track priority = 20), so that all traffic destined to the Internet is sent through Switch 1 instead.

ARP behavior with VRRP-E

In the VRRP-E implementation, the source MAC address of the gratuitous Address Resolution Protocol (ARP) request sent by the VRRP-E Master router is the VRRP-E virtual MAC address. When the router (either the Master or Backup router) sends an ARP request or reply packet, the sender's MAC address becomes the MAC address of the interface on the router. When an ARP request packet for the virtual router IP address is received by the Backup router, it is forwarded to the Master router to resolve the ARP request. Only the Master router answers the ARP request for the virtual router IP address.

Comparison of VRRP and VRRP-E

This section compares router redundancy protocols.

VRRP

VRRP is a standards-based protocol, described in RFC 2338. The Brocade implementation of VRRP contains the features in RFC 2338. The Brocade implementation also provides the following additional features:

- Track ports – A Brocade feature that enables you to diagnose the health of all the Layer 3 switch ports used by the backed-up VRID, instead of only the port connected to the client subnet. Refer to "Track ports and track priority" on page 416.
- Suppression of RIP advertisements on Backup routers for the backed-up interface – You can enable the Layer 3 switches to advertise only the path to the Master router for the backed-up interface. Normally, a VRRP Backup router includes route information for the interface it is backing up in RIP advertisements.

Brocade Layer 3 switches configured for VRRP can interoperate with third-party routers using VRRP.

VRRP-E

VRRP-E is a Brocade protocol that provides the benefits of VRRP without the limitations. VRRP-E is unlike VRRP in the following ways:

- There is no "Owner" router. You do not need to use an IP address configured on one of the Layer 3 switches as the virtual router ID (VRID), which is the address you are backing up for redundancy. The VRID is independent of the IP interfaces configured in the Layer 3 switches. As a result, the protocol does not have an "Owner" as VRRP does.
- There is no restriction on which router can be the default Master router. In VRRP, the "Owner" (the Layer 3 switch on which the IP interface that is used for the VRID is configured) must be the default Master.

Brocade Layer 3 switches configured for VRRP-E can interoperate only with other Brocade Layer 3 switches.

Architectural differences between VRRP and VRRP-E

The protocols have the following architectural differences.

Management protocol

- VRRP – VRRP routers send VRRP Hello and Hello messages to IP Multicast address 224.0.0.18.
- VRRP-E – VRRP-E sends messages to destination MAC address 01-00-5E-00-00-02 and destination IP address 224.0.0.2 (the standard IP multicast address for “all routers”).

Virtual router IP address (the address you are backing up)

- VRRP – The virtual router IP address is the same as an IP address or virtual interface configured on one of the Layer 3 switches, which is the “Owner” and becomes the default Master.
- VRRP-E – The virtual router IP address is the gateway address you want to back up, but does not need to be an IP interface configured on one of the Layer 3 switch ports or a virtual interface.

Master and Backup routers

- VRRP – The “Owner” of the IP address of the VRID is the default Master and has the highest priority (255). The precedence of the Backup routers is determined by their priorities. The default Master is always the Owner of the IP address of the VRID.
- VRRP-E – The Master and Backup routers are selected based on their priority. You can configure any of the Layer 3 switches to be the Master by giving it the highest priority. There is no Owner.

VRRP and VRRP-E parameters

Table 79 lists the VRRP and VRRP-E parameters. Most of the parameters and default values are the same for both protocols. The exceptions are noted in the table.

TABLE 79 VRRP and VRRP-E parameters

Parameter	Description	Default	For more information
Protocol	The Virtual Router Redundancy Protocol (VRRP) based on RFC 2338 or VRRP-Extended, the Brocade-enhanced implementation of VRRP.	Disabled **NOTE:** Only one of the protocols can be enabled at a time.	page 425 page 430
VRRP or VRRP-E router	The Brocade Layer 3 switch active participation as a VRRP or VRRP-E router. Enabling the protocol does not activate the Layer 3 switch for VRRP or VRRP-E. You must activate the switch as a VRRP or VRRP-E router after you configure the VRRP or VRRP-E parameters.	Inactive	page 425 page 430
Virtual Router ID (VRID)	The ID of the virtual router you are creating by configuring multiple routers to back up an IP interface. You must configure the same VRID on each router that you want to use to back up the address.	None	page 414 page 425 page 430
Virtual Router IP address	This is the address you are backing up. • VRRP – The virtual router IP address must be a real IP address configured on the VRID interface on one of the VRRP routers. This router is the IP address Owner and is the default Master. • VRRP-E – The virtual router IP address must be in the same subnet as a real IP address configured on the VRRP-E interface, but cannot be the same as a real IP address configured on the interface.	None	page 414 page 425 page 430
VRID MAC address	The source MAC address in VRRP or VRRP-E packets sent from the VRID interface, and the destination for packets sent to the VRID: • VRRP – A virtual MAC address defined as 00-00-00-00-01-*vrid* for IPv4 VRRP, and 00-00-00-00-02-*vrid* for VRRP v3. The Master owns the virtual MAC address. • VRRP-E – A virtual MAC address defined as 00-00-00-*hash-value-vrid* for IPv4 VRRP-E and IPv6 VRRP-E, where *hash-value* is a two-octet hashed value for the IP address and *vrid* is the ID of the virtual router.	Not configurable	page 414

TABLE 79 VRRP and VRRP-E parameters (Continued)

Parameter	Description	Default	For more information
Authentication type	The type of authentication the VRRP or VRRP-E interfaces use to validate VRRP or VRRP-E packets. • No authentication – The interfaces do not use authentication. This is the VRRP default. • Simple – The interface uses a simple text-string as a password in packets sent on the interface. If the interface uses simple password authentication, the VRID configured on the interface must use the same authentication type and the same password. • HMAC-MD5-96 (VRRP-E only) – The interface uses HMAC-MD5-96 authentication for VRRP-E packets. NOTE: HMAC-MD5-96 authentication is only supported for IPv4 or IPv6 VRRP-E. HMAC-MD5-96 is not supported by VRRP. Authentication is not supported for VRRP v3.	No authentication	page 416 page 433
Router type	Whether the router is an Owner or a Backup. • Owner (VRRP only) – The router on which the real IP address used by the VRID is configured. • Backup – Routers that can provide routing services for the VRID but do not have a real IP address matching the VRID.	VRRP – The Owner is always the router that has the real IP address used by the VRID. All other routers for the VRID are Backups. VRRP-E – All routers for the VRID are Backups.	page 435
Backup priority	A numeric value that determines a Backup router's preferability for becoming the Master for the VRID. During negotiation, the router with the highest priority becomes the Master. • VRRP – The Owner has the highest priority (255); other routers (backups) can have a priority from 3 through 254. • VRRP-E – All routers are Backups and have the same priority by default. If two or more Backups are tied with the highest priority, the Backup interface with the highest IP address becomes the Master for the VRID.	VRRP v2 and IPv6 VRRP v3 - The value is 255 for the Owner and 100 for the Backups. VRRP-E v2 and IPv6 VRRP-E v3 -The value is 100 for all Backups.	page 435
Suppression of RIP advertisements	A router that is running RIP normally advertises routes to a backed-up VRID even when the router is not currently the active router for the VRID. Suppression of these advertisements helps ensure that other routers do not receive invalid route paths for the VRID. NOTE: Suppression of RIP advertisements is not supported for VRRP v3 and VRRP-E v3.	Disabled	page 436
Hello interval	The number of seconds or milliseconds between Hello messages from the Master to the Backups for a given VRID. The interval can be from 1 through 84 seconds for VRRP v2, VRRP-E v2, and IPv6 VRRP-E. The interval for VRRP v3 can be from 100 through 8400 milliseconds.	One second (VRRP v2 and VRRP-E v2, and IPv6 VRRP-E) 1000 milliseconds (VRRP v3).	page 415 page 437

TABLE 79 VRRP and VRRP-E parameters (Continued)

Parameter	Description	Default	For more information
Dead interval	The number of seconds or milliseconds a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID.	The dead interval calculation for VRRP v6 or VRRP-E v6 is: Dead Interval: (3 X Hello Interval) + Skew Time Skew Time is 256 - Priority X Hello Interval / 256. For VRRP v3, the default is 3600 milliseconds. The configurable range is from 100 through 8400 milliseconds. For VRRP-E v3, the default is 3600 milliseconds. The configurable range is from 1 through 84 seconds.	page 415 page 438
Backup Hello interval	The number of seconds between Hello messages from a Backup to the Master. The message interval can be from 60 through 3600 seconds. You must enable the Backup to send the messages. The messages are disabled by default on Backups. The current Master (whether the VRRP Owner or a Backup) sends Hello messages by default.	Disabled 60 seconds when enabled	page 415 page 438
Track port	Another Layer 3 switch port or virtual interface whose link status is tracked by the VRID interface. If the link for a tracked interface goes down, the VRRP or VRRP-E priority of the VRID interface is changed, causing the devices to renegotiate for the Master. **NOTE:** Track port is not supported by VRRP v3.	None	page 416 page 439
Track priority	A VRRP or VRRP-E priority value assigned to the tracked ports. If a tracked port link goes down, the VRID port VRRP or VRRP-E priority changes: • VRRP – The priority changes to the value of the tracked port priority. • VRRP-E – The VRID port priority is reduced by the amount of the tracked port priority. **NOTE:** Track priority is not supported by VRRP v3.	VRRP – 2 VRRP-E – 5	page 416 page 439
Backup preempt mode	Prevents a Backup with a higher VRRP priority from taking control of the VRID from another Backup that has a lower priority but has already assumed control of the VRID.	Enabled	page 440
Timer scale	Adjusts the timers for the Hello interval, Dead interval, Backup Hello interval, and Hold-down interval. **NOTE:** The timer scale is not supported for IPv6 VRRP v3.	1	page 440

TABLE 79 VRRP and VRRP-E parameters (Continued)

Parameter	Description	Default	For more information
VRRP-E slow start timer	Causes a specified amount of time to elapse between the time the original Master is restored and when it takes over from the Backup. This interval allows time for OSPF convergence when the Master is restored. For VRRP-E only.	Disabled	page 441
Short-path forwarding	Enables VRRP-E extension for server virtualization. If enabled, the traffic that is destined to the clients travels through the short-path forwarding path to reach the client (as shown in Figure 33 on page 443). Any packets coming from the local subnet of the virtual IP address are forwarded either by the VRRP-E master router or VRRP-E backup router depending on which router received the packets.	Disabled	page 436

Note regarding disabling VRRP or VRRP-E

If you disable VRRP or VRRP-E, the Layer 3 switch removes all the configuration information for the disabled protocol from the running-config file. Moreover, when you save the configuration to the startup-config file after disabling one of these protocols, all the configuration information for the disabled protocol is removed from the startup-config file.

The CLI displays a warning message such as the following.

```
Brocade Router1(config-vrrp-router)#no router vrrp
router vrrp mode now disabled. All vrrp config data will be lost when writing to
flash!
```

If you have disabled the protocol but have not yet saved the configuration to the startup-config file and reloaded the software, you can restore the configuration information by re-entering the command to enable the protocol (for example, **router vrrp**). If you have already saved the configuration to the startup-config file and reloaded the software, the information is gone.

If you are testing a VRRP or VRRP-E configuration and are likely to disable and re-enable the protocol, you may want to make a backup copy of the startup-config file containing the protocol configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

Basic VRRP parameter configuration

To implement a simple VRRP configuration using all the default values, enter the commands shown in the following sections.

Configuration rules for VRRP

- The interfaces of all routers in a VRID must be in the same IP subnet.

- The IP addresses associated with the VRID must already be configured on the router that will be the Owner.
- An IP address associated with the VRID must be on only one router.
- The Hello interval must be set to the same value on the Owner and Backup routers for the VRID.
- The dead interval must be set to the same value on the Owner and Backup routers for the VRID.
- The track priority on a router must be lower than the router VRRP priority. Also, the track priority on the Owner must be higher than the track priority on the Backup routers.

NOTE
When you use the **router vrrp** command or the **ipv6 router vrrp** command to enter the VRRP configuration mode, the command prompt does not change and results in the following general configuration command prompt: Brocade`(config)#`. This differs from entering the VRRP extended mode, where entering the **router vrrp-extended** command results in a command prompt such as the following: `(config-VRRP-E-router)#.` For IPv6 VRRP extended mode, when entering the **ipv6 router vrrp-extended** command, this results in a command prompt such as the following: `(config-ipv6-VRRP-E-router)#.`

Configuring the Owner for IPv4 VRRP

To configure the VRRP Owner router for IPv4, enter the following commands on the router.

```
Brocade Router1(config)#router vrrp
Brocade Router1(config)#interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)#ip-address 192.168.5.1
Brocade Router1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#owner
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#ip-address 192.168.5.1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#activate
```

Syntax: **[no] router vrrp**

Syntax: **[no] ip-address** *ip-addr*

Syntax: **[no] ip vrrp vrid** *num*

Syntax: **[no] owner [track-priority** *value***]**

Syntax: **[no] activate**

The *ip-addr* variable specifies the IPv4 address of the Owner router.

The IP address you assign to the Owner must be an IP address configured on an interface that belongs to the virtual router.

The *num* variable specifies the virtual router ID.

The **track-priority** *value* option changes the track-port priority for this interface and the VRID from the default (2) to a value from 1 through the maximum VRID supported by the device.

Configuring the Owner for IPv6 VRRP

To configure the VRRP Owner router for IPv6, enter the following commands on the router.

NOTE
You must first configure the **ipv6 unicast-routing** command at the global configuration level to enable IPv6 VRRP on the router.

```
Brocade Router1(config)# ipv6 unicast-routing
Brocade Router1(config)# ipv6 router vrrp
Brocade Router1(config)# interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)# ipv6-address 3013::1/64
Brocade Router1(config-if-e10000-1/1/6)# ipv6 vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)# owner
Brocade Router1(config-if-e10000-1/1/6-vrid-1)# ipv6-address 3013::1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
```

Syntax: [no] **ipv6 unicast-routing**

Syntax: [no] **ipv6 router vrrp**

Syntax: [no] **ipv6-address** *ipv6-address*

Syntax: [no] **ipv6 vrrp vrid** *num*

Syntax: [no] **owner**

Syntax: [no] **activate**

The *num* variable specifies the virtual router ID.

The *ipv6-addr* variable specifies the IPv6 address of the Owner router.

The IP address you assign to the Owner must be an IP address configured on an interface that belongs to the virtual router.

The **ipv6 router vrrp** command enables IPv6 VRRP v3 routing on the interface. All IPv6 VRRP router instances for a VRID are also enabled on the interface.

When the **no ipv6 router vrrp** command is enabled, all IPv6 VRRP router instances for a specific VRID are deleted from the interface, and the running configuration is lost when writing to flash. You must enable the **write memory** command to save your configuration. The following message is displayed when the **no ipv6 router vrrp** command is enabled.

```
Router1(config)#no ipv6 router vrrp
ipv6 router vrrp is disabled. All vrrp (ipv6) config data will be lost when
writing to flash!!
```

Configuring a Backup for IPv4 VRRP

To configure the VRRP Backup router for IPv4, enter the following commands.

```
Brocade Router2(config)#router vrrp
Brocade Router2(config)#interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)#ip-address 192.168.5.3
Brocade Router2(config-if-e10000-1/1/5)#ip vrrp vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#advertise backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#ip-address 192.168.5.1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#activate
VRRP router 2 for interface is activating
```

Syntax: [no] **router vrrp**

Syntax: [**no**] **ip-address** *ip-addr*

Syntax: [**no**] **ip vrrp vrid** *num*

Syntax: [**no**] **backup** [**priority** *value*] [**track-priority** *value*]

Syntax: [**no**] **advertise backup**

Syntax: [**no**] **activate**

When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same subnet as the address associated with the VRID by the Owner. However, the address cannot be the same.

The *num* variable specifies the virtual router ID.

The **priority** *value* option specifies the VRRP priority for this virtual router. You can specify a value from 3 through 254. The default is 100.

The **track-priority** *value* option specifies that VRRP monitors the state of the interface. You can specify a value from 3 through 254. The default is 100.

By default, Backup routers do not send Hello messages to advertise themselves to the Master. The **advertise backup** command is used to enable a Backup router to send Hello messages to the Master.

Configuring a Backup for IPv6 VRRP

To configure the VRRP Backup router for IPv6, enter the following commands.

```
Brocade Router2(config)# ipv6 router vrrp
Brocade Router2(config)# interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)# ipv6-address 2001:db8::3/64
Brocade Router2(config-if-e10000-1/1/5)# ipv6 vrrp vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# advertise backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# ipv6-address 2001:db8::1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# activate
```

When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same subnet as the address associated with the VRID by the Owner. However, the address cannot be the same.

Syntax: [**no**] **ipv6 router vrrp**

Syntax: [**no**] **ipv6-address** *ipv6-addr*

Syntax: [**no**] **ipv6 vrrp vrid** *num*

Syntax: [**no**] **backup** [**priority** *value*]

Syntax: [**no**] **advertise backup**

Syntax: [**no**] **activate**

The *ipv6-addr* variable specifies the IPv6 address of the Backup router.

The *num* variable specifies the virtual router ID.

The **priority** *value* option specifies the IPv6 VRRP priority for this virtual Backup router. You can specify a value from 3 through 254. The default is 100.

By default, Backup routers do not send Hello messages to advertise themselves to the Master. The **advertise backup** command is used to enable a Backup router to send Hello messages to the Master.

Configuration considerations for IPv6 VRRP v3 and IPv6 VRRP-E v3 support on Brocade devices

Consider the following when enabling IPv6 VRRP v3 mode and IPv6 VRRP-E v3 mode on devices:

- You can configure only one protocol (Layer 3 VSRP, VRRP, or VRRP-E) on a router at a single time. However, VRRP or VRRP-E can be configured with IPv4 and IPv6 concurrently on a router.
- Scale timer configuration does not affect timer values, nor does it scale timer values for virtual routers configured with sub-second time values for IPv6 VRRP and IPv4 VRRP v3 modes.
- Abdication of a VRRP Owner router in an IPv6 environment is not supported. Abdication of an Owner router is a Brocade-specific enhancement to VRRP. Abdication of an Owner router is possible by changing the Owner's priority, or by configuring track ports for an Owner router.
  - For IPv6 VRRP v3 only, the tracking port configuration is not allowed if the router is configured as the VRRP Owner. This conforms to RFC 5798.
  - For the IPv6 VRRP v3 Owner router only, the priority configuration is not allowed. The Owner router priority is always 255. This conforms to RFC 5798.
- Hitless switchover is not supported for IPv6 VRRP and IPv6 VRRP-E environments.
- Interoperability is not supported for a VRID when VRRP routers are configured as VRRP v2 or v3.
- Brocade does not recommend that you re-use the same VRID across IPv6 VRRP-E and IPv4 VRRP-E if they are in the same broadcast domain.
- There is no specified restriction for configuring VRRP or VRRP-E instances if they are within the maximum VRID range. The maximum number of supported VRRP or VRRP-E router instances is 254 for IPv4 environments. The maximum number of supported VRRP or VRRP-E router instances is 128 for IPv6 environments.

Basic VRRP-E parameter configuration

The following sections describe the configuration of the parameters specific to IPv4 and IPv6 VRRP-E.

Configuration rules for VRRP-E

Consider the following rules when configuring VRRP-E:

- The interfaces of all routers in a VRID must be in the same IP subnet.
- The IP address associated with the VRID cannot be configured on any of the Layer 3 switches.
- The Hello interval must be set to the same value on all the Layer 3 switches.
- The dead interval must be set to the same value on all the Layer 3 switches.
- The track priority for a VRID must be lower than the VRRP-E priority.

Configuring IPv4 VRRP-E

VRRP-E is configured at the interface level. To implement a simple IPv4 VRRP-E configuration using all the default values, enter commands such as the following on each Layer 3 switch.

```
Brocade Router2(config)#router vrrp-extended
Brocade Router2(config)#interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)#ip-address 192.168.5.3
Brocade Router2(config-if-e10000-1/1/5)#ip vrrp-extended vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#advertise backup
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#ip-address 192.168.5.254
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#activate
```

Syntax: [no] **router vrrp-extended**

Syntax: [no] **ip-address** *ip-address*

Syntax: [no] **ip vrrp-extended vrid** *vrid*

Syntax: [no] **backup** [**priority** *value*] [**track-priority** *value*]

Syntax: [no] **advertise backup**

Syntax: [no] **activate**

The *vrid* variable specifies the virtual router ID.

The *ip-addr* variable specifies the IPv4 address of the router.

You must identify a VRRP-E router as a Backup before you can activate the virtual router on a Brocade device. However, after you configure the virtual router, you can use the **backup** command to change its priority or track priority.

The **priority** *value* option specifies the IPv4 VRRP-E priority for this virtual Backup router. You can specify a value from 3 through 254. The default is 100.

The **track-priority** *value* option changes the track port priority of a Backup router. You can specify a value from 1 through 254. The default is 2.

NOTE
You also can use the **enable** command to activate the configuration. This command does the same thing as the **activate** command.

Configuring IPv6 VRRP-E

To implement an IPv6 VRRP-E configuration using all the default values, enter the following commands.

NOTE
You must first configure the **ipv6 unicast-routing** command at the global configuration level to enable IPv6 VRRP-E on the router.

```
Brocade Router2(config)# ipv6 unicast-routing
Brocade Router2(config)# ipv6 router vrrp-extended
Brocade Router2(config-ipv6-VRRP-E-router)# interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)# ipv6-address 2001:db8::2/64
Brocade Router2(config-if-e10000-1/1/5)# ipv6 vrrp-extended vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# backup priority 50 track-priority
10
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# ipv6-address 2001:db8::99
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# activate
```

Syntax: [**no**] **ipv6 unicast-routing**

Syntax: **ipv6 router vrrp-extended**

Syntax: [**no**] **ipv6-address** *ipv6-addr*

Syntax: **ipv6 vrrp-extended vrid** *vrid*

Syntax: [**no**] **backup** [**priority** *value*] [**track-priority** *value*]

Syntax: [**no**] **activate**

The *vrid* variable specifies the virtual router ID.

The *ipv6-addr* variable specifies the IPv6 address of the router.

You must identify a VRRP-E router as a Backup before you can activate the virtual router on a Brocade device. However, after you configure the virtual router, you can use the **backup** command to change its priority or track priority.

The **priority** *value* option specifies the IPv6 VRRP-E priority for this virtual Backup router. You can specify a value from 3 through 254. The default is 100.

The **track-priority** *value* option changes the track port priority of a Backup router. You can specify a value from 1 through 254. The default is 2.

NOTE
You also can use the **enable** command to activate the configuration. This command does the same thing as the **activate** command.

When the **no ipv6 router vrrp-extended** command is enabled, all IPv6 VRRP-E instances for a specific VRID are deleted from the interface, and the running configuration is lost when writing to flash. You must enable the **write memory** command to save your configuration. The following message is displayed when the **no ipv6 router vrrp-extended** command is enabled.

```
Brocade Router2(config)#no ipv6 router vrrp-extended
ipv6 router VRRP-E is disabled. All VRRP-E (ipv6) config data will be lost when
writing to flash!!
```

Additional VRRP and VRRP-E parameter configuration

You can modify the following VRRP and VRRP-E parameters on an individual VRID basis. These parameters apply to both protocols:

- Authentication type (if the interfaces on which you configure the VRID use authentication)
- Router type (Owner or Backup)

NOTE
For VRRP, change the router type only if you have moved the real IP address from one router to another or you accidentally configured the IP address Owner as a Backup.

For VRRP-E, the router type is always Backup. You cannot change the type to Owner.

- Suppression of RIP advertisements on Backup routes for the backed-up interface
- Hello interval
- Dead interval
- Backup Hello messages and message timer (Backup advertisement)
- Track port
- Track priority
- Backup preempt mode
- Timer scale
- VRRP-E slow start timer

Refer to "VRRP and VRRP-E parameters" on page 422 for a summary of the parameters and their defaults.

VRRP and VRRP-E authentication types

This section describes VRRP and VRRP-E authentication parameters.

Configuring authentication type

The Brocade implementation of VRRP and VRRP-E supports the following authentication types for authenticating VRRP and VRRP-E traffic:

- No authentication – The interfaces do not use authentication. This is the default for VRRP and VRRP-E.
- Simple – The interfaces use a simple text-string as a password in packets sent on the interface. If the interfaces use simple password authentication, the VRID configured on the interfaces must use the same authentication type and the same password.

VRRP-E also supports the following authentication type:

- HMAC-MD5-96 – The interfaces use HMAC-MD5-96 authentication for VRRP-E packets.

NOTE
HMAC-MD5-96 authentication is not supported for VRRP.

To configure the VRID interface on Switch 1 for simple password authentication using the password “ourpword”, enter the following commands.

Configuring Switch 1

```
Brocade Switch1(config)#inter e 1/1/6
Brocade Switch1(config-if-e10000-1/1/6)#ip vrrp auth-type simple-text-auth
ourpword
```

VRRP syntax

Syntax: **auth-type no-auth** | **simple-text-auth** *auth-data*

The **auth-type no-auth** option indicates that the VRID and the interface it is configured on do not use authentication.

The **simple-text-auth** *auth-data* option indicates that the VRID and the interface it is configured on use a simple text password for authentication. The *auth-data* variable is the password. If you use this variable, make sure all interfaces on all the routers supporting this VRID are configured for simple password authentication and use the same password.

NOTE
For VRRP v3, authentication is deprecated by RFC 5768.

VRRP-E syntax

For IPv4 VRRP-E:

Syntax: ip vrrp-extended auth-type no-auth | **simple-text-auth** *auth-data* | **md5-auth** [0 |1] *key*

For IPv6 VRRP-E:

Syntax: ipv6 vrrp-extended auth-type no-auth | **simple-text-auth** *auth-data* | **md5-auth** [0 |1] *key*

The values for the **no-auth** and **simple-text-auth** *auth-data* options are the same as for VRRP.

The **md5-auth** option configures the interface to use HMAC-MD5-96 for VRRP-E authentication.

The *key* variable is the MD5 encryption key, which can be up to 64 characters long. The optional [0 |1] parameter configures whether the MD5 password is encrypted, as follows:

- If you do not enter this parameter and enter the key as clear text, the key appears encrypted in the device configuration and command outputs.
- If you enter 0 and enter the key as clear text, the key appears as clear text in the device configuration and command outputs.
- If you enter 1 and enter the key in encrypted format, the key appears in encrypted format in the device configuration and command outputs.

Syslog messages for VRRP-E HMAC-MD5-96 authentication

If an interface is configured with HMAC-MD5-96 authentication, all VRRP-E packets received on this interface are authenticated with the HMAC-MD5-96 algorithm using the shared secret key configured on the interface.

If a packet is received that fails this HMAC-MD5-96 authentication check, the packet gets dropped. Additionally, if syslog is enabled, a syslog message is generated to notify the administrator about an authentication failure. The message includes the VRID received in the packet's VRRP message and the interface on which the packet was received. These syslog messages will be rate limited to 20 log messages within a span of 5 minutes, starting from the first packet received that fails the HMAC-MD5-96 authentication check.

VRRP router type

A VRRP interface is either an Owner or a Backup router for a given VRID. By default, the Owner becomes the Master. A Backup router becomes the Master only if the Master becomes unavailable.

A VRRP-E interface is always a Backup router for its VRID. The Backup router with the highest VRRP priority becomes the Master.

This section describes how to specify the interface type, how to change the type for VRRP, and how to set or change the interface VRRP or VRRP-E priority and track priority for the VRID.

NOTE
You can force a VRRP Master router to abdicate (give away control) of the VRID to a Backup router by temporarily changing the Master VRRP priority to a value less than the Backup. Refer to "Forcing a Master router to abdicate to a Backup router" on page 445.

NOTE
The Owner type is not applicable to VRRP-E.

NOTE
The IP addresses you associate with the Owner must be real IP addresses on the interface on which you configure the VRID.

When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same subnet as the address associated with the VRID by the Owner. However, the address cannot be the same.

Configuring Router 1 as VRRP VRID Owner

To configure Router1 as a VRRP VRID Owner, enter the following commands.

```
Brocade Router1(config)#interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#owner
```

Configuring Router 2 as VRRP Backup

To configure Router2 as a VRRP Backup for the same VRID, and set its VRRP priority, enter the following commands.

```
Brocade Router2(config)#interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)#ip vrrp vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#backup priority 120
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#advertise backup
```

Configuring an IPv6 VRRP v3 interface as a Backup for a VRID

To configure an IPv6 VRRP v3 interface as a Backup for a VRID, and set its VRRP priority and track priority, enter commands such as the following.

```
Brocade Router2(config)# interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)# ipv6 vrrp vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)# backup priority 50 track-priority
10
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#advertise backup
```

Configuring an IPv6 VRRP-E v3 interface as a Backup for a VRID

To configure an IPv6 VRRP-E v3 interface as a Backup for a VRID, and set its VRRP-E priority and track priority, enter commands such as the following.

```
Brocade Router2(config)#interface ethernet 1/1/1
Brocade Router2(config-if-e10000-1/1/1)#ipv6 vrrp-extended vrid 1
Brocade Router2(config-if-e10000-1/1/1-vrid-1)#backup priority 50 track-priority
10
Brocade Router2(config-if-e10000-1/1/1-vrid-1)#advertise backup
```

VRRP v2 and IPv6 VRRP v3 syntax

Syntax: **owner** [**track-priority** *value*]

The **track-priority** *value* option changes the track port priority for this interface and VRID from the default (2) to a value from 1 through 254.

Syntax: **backup** [**priority** *value*] [**track-priority** *value*]

The **priority** *value* option specifies the VRRP priority for this interface and VRID. You can specify a value from 3 through 254. The default is 100.

The **track-priority** *value* option is the same as with the **owner** [**track-priority** *value*] command.

VRRP-E v2 and IPv6 VRRP-E v3 syntax

Syntax: **backup** [**priority** *value*] [**track-priority** *value*]

The software requires you to identify a VRRP-E interface as a Backup for its VRID before you can activate the interface for the VRID. However, after you configure the VRID, you can use this command to change its priority or track priority. The option values are the same as for VRRP.

Suppression of RIP advertisements

NOTE
Suppression of RIPng advertisements on Backup routers for the backup interface is not supported by IPv6 VRRP v3 and IPv6 VRRP-E v3.

Normally, a VRRP or VRRP-E Backup includes route information for the virtual IP address (the backed-up interface) in RIP advertisements. As a result, other routers receive multiple paths for the backed-up interface and might sometimes unsuccessfully use the path to the Backup router rather than the path to the Master.

You can prevent the Backup routers from advertising route information for the backed-up interface by enabling suppression of the advertisements.

Suppressing RIP advertisements for the backed-up interface in Router 2

To suppress RIP advertisements for the backed-up interface in Router 2, enter the following commands.

```
Brocade Router2(config)#router rip
Brocade Router2(config-rip-router)#use-vrrp-path
```

Syntax: use-vrrp-path

The syntax is the same for VRRP and VRRP-E.

Hello interval configuration

The Master periodically sends Hello messages to the Backup routers. The Backup routers use the Hello messages as verification that the Master is still online. If the Backup routers stop receiving the Hello messages for the period of time specified by the dead interval, the Backup routers determine that the Master router is dead. At this point, the Backup router with the highest priority becomes the new Master router.

NOTE
The default dead interval is three times the Hello interval plus the Skew time. Generally, if you change the Hello interval, you also should change the dead interval on the Backup routers.

To change the Hello interval on the Master to 10 seconds, enter the following commands.

```
Brocade Router1(config)#interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#hello-interval 10
```

Syntax: [no] hello-interval *seconds*

The *seconds* variable specifies the Hello interval value from 1 through 84 seconds for VRRP v2, VRRP-E v2, and IPv6 VRRP-E. The default is 1 second.

To change the Hello interval on the Master to 200 milliseconds for IPv6 VRRP v3, enter the following commands.

```
Brocade Router1(config)# interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)# ipv6 vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)# hello-interval 200
```

Syntax: [no] hello-interval *milliseconds*

The *milliseconds* variable can be from 100 through 8400 milliseconds. The default is 1000 milliseconds.

Dead interval configuration

The dead interval is the number of seconds a Backup router waits for a Hello message from the Master before determining that the Master is dead. When Backup routers determine that the Master is dead, the Backup with the highest priority becomes the new Master.

If the value for the dead interval is not configured, then the current dead interval is equal to three times the Hello interval plus the Skew time (where Skew time is equal to (256 - priority) divided by 256).

To change the dead interval on a Backup to 30 seconds, enter the following commands.

```
Brocade Router2(config)#interface ethernet 1/1/5
Brocade Router2(config-if-e10000-1/1/5)#ip vrrp vrid 1
Brocade Router2(config-if-e10000-1/1/5-vrid-1)#dead-interval 30
```

Syntax: dead-interval *value*

The *value* variable is from 1 through 84 seconds for VRRP v2 and VRRP-E v2. For other versions, the *value* variable is from 100 through 8400 milliseconds. The default is 3600 milliseconds.

NOTE
If the **dead-interval** command is not configured on a VRRP v3 interface, then a zero value is displayed in the output of the **show ipv6 VRRP-Extended** command.

Backup Hello message state and interval

NOTE
The **advertise backup** command is supported by IPv4 VRRP v2, and IPv6 VRRP v3 and IPv6 VRRP-E v3.

By default, Backup routers do not send Hello messages to advertise themselves to the Master. You can enable these messages if desired and also change the message interval.

To enable a Backup router to send Hello messages to the Master, enter the following commands.

```
Brocade(config)#router vrrp
Brocade(config)#interface ethernet 1/1/6
Brocade(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade(config-if-e10000-1/1/6-vrid-1)#advertise backup
```

Syntax: [**no**] **advertise backup**

When you enable a Backup to send Hello messages, the Backup sends a Hello message to the Master every 60 seconds by default. You can change the interval to be up to 3600 seconds. To change the Hello message interval, enter the following commands.

```
Brocade(config)#router vrrp
Brocade(config)#interface ethernet 1/1/6
Brocade(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade(config-if-e10000-1/1/6-vrid-1)#backup-hello-interval 180
```

Syntax: [**no**] **backup-hello-interval** *num*

The *num* variable specifies the message interval and can be from 60 through 3600 seconds. The default is 60 seconds.

The syntax is the same for VRRP v2 and IPv6 VRRP v3, and VRRP-E v2 and IPv6 VRRP-E v3.

Track port configuration

NOTE
Track port is not supported by VRRP v3.

You can configure the VRID on one interface to track the link state of another interface on the Layer 3 switch. This capability is quite useful for tracking the state of the exit interface for the path for which the VRID is providing redundancy. Refer to "Track ports and track priority" on page 416.

To configure interface 1/1/6 on Router 1 to track interface 1/1/2, enter the following commands.

```
Brocade Router1(config)#interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#track-port ethernet 1/1/2
```

Syntax: **track-port ethernet** *stack-unit/slotnum/portnum* | **ve** *num*

The syntax is the same for VRRP and VRRP-E.

Track priority configuration

NOTE
Track priority is not supported by IPv6 VRRP v3.

When you configure a VRID to track the link state of other interfaces, and one of the tracked interfaces goes down, the software changes the VRRP or VRRP-E priority of the VRID interface:

- For VRRP, the software changes the priority of the VRID to the track priority, which typically is lower than the VRID priority and lower than the VRID priorities configured on the Backups. For example, if the VRRP interface priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VRRP interface priority to 60.
- For VRRP-E, the software reduces the VRID priority by the amount of the priority of the tracked interface that went down. For example, if the VRRP-E interface priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VRRP-E interface priority to 40. If another tracked interface goes down, the software reduces the VRID priority again, by the amount of the tracked interface track priority.

The default track priority for an Owner for VRRP v2, IPv6 VRRP v3, and VRRP-E v2 and IPv6 VRRP-E v3 is 2. The default track priority for Backup routers is 1.

You enter the track priority as a value with the **owner** or **backup** command. Refer to "Track port configuration" on page 439.

Syntax: **owner** [**track-priority** *value*]

Syntax: **backup** [**priority** *value*] [**track-priority** *value*]

The syntax is the same for VRRP and VRRP-E.

Backup preempt configuration

By default, a Backup that has a higher priority than another Backup that has become the Master can preempt the Master, and take over the role of Master. If you want to prevent this behavior, disable preemption.

Preemption applies only to Backups and takes effect only when the Master has failed and a Backup has assumed ownership of the VRID. The feature prevents a Backup with a higher priority from taking over as Master from another Backup that has a lower priority but has already become the Master of the VRID.

Preemption is especially useful for preventing flapping in situations where there are multiple Backups and a Backup with a lower priority than another Backup has assumed ownership, because the Backup with the higher priority was unavailable when ownership changed.

If you enable the non-preempt mode (thus disabling the preemption feature) on all the Backups, the Backup that becomes the Master following the disappearance of the Master continues to be the Master. The new Master is not preempted.

NOTE
In VRRP, regardless of the setting for the preempt parameter, the Owner always becomes the Master again when it comes back online.

To disable preemption on a Backup, enter commands such as the following.

```
Brocade Router1(config)#interface ethernet 1/1/6
Brocade Router1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Router1(config-if-e10000-1/1/6-vrid-1)#non-preempt-mode
```

Syntax: [no] non-preempt-mode

The syntax is the same for VRRP and VRRP-E.

Changing the timer scale

NOTE
Changing the timer scale is supported for IPv4 VRRP v2, IPv4 VRRP-E v2, and IPv6 VRRP-E v3. It is not supported for IPv6 VRRP v3.

To achieve sub-second failover times, you can shorten the duration of all scale timers for VSRP, VRRP, and VRRP-E by adjusting the timer scale. The timer scale is a value used by the software to calculate the timers. By default, the scale value is 1. If you increase the timer scale, each timer's value is divided by the scale value. Using the timer scale to adjust timer values enables you to easily change all the timers while preserving the ratios among their values. Table 80 shows timer scale values.

TABLE 80 Time scale values

Timer	Timer scale	Timer value
Hello interval	1	1 second
	2	0.5 seconds
Dead interval	1	3 seconds
	2	1.5 seconds

TABLE 80 Time scale values (Continued)

Timer	Timer scale	Timer value
Backup Hello interval	1	60 seconds
	2	30 seconds
Hold-down interval	1	2 seconds
	2	1 second

If you configure the device to receive its timer values from the Master, the Backup also receives the timer scale value from the Master.

To change the timer scale, enter a command such as the following at the global CONFIG level of the CLI.

```
Brocade(config)# scale-timer 2
```

This command changes the scale to 2. All VSRP, VRRP, and VRRP-E timer values will be divided by 2.

Syntax: [no] scale-timer *num*

The *num* variable specifies the multiplier. You can specify a timer scale from 1 through 10. However, Brocade recommends the timer scale of 1 or 2 for VRRP and VRRP-E.

NOTE
Be cautious when configuring the **scale-timer** command in a VRRP or VRRP-E scaled environment. VSRP, VRRP, and VRRP-E are time-sensitive protocols and system behavior cannot be predicted when the timers are scaled.

VRRP-E slow start timer

In a VRRP-E configuration, if a Master router goes down, the Backup router with the highest priority takes over after expiration of the dead interval. When the original Master router comes back up again, it takes over from the Backup router (which became the Master router when the original Master router went down). By default, this transition from Backup back to Master takes place immediately. However, you can configure the VRRP-E slow start timer feature, which causes a specified amount of time to elapse between the time the Master is restored and when it takes over from the Backup. This interval allows time for OSPF convergence when the Master is restored.

To set the IPv4 VRRP-E slow start timer to 30 seconds, enter the following commands.

```
Brocade(config)#router vrrp-extended
Brocade(config-VRRP-E-router)#slow-start 30
```

To set the IPv6 VRRP-E slow start timer to 60 seconds, enter the following commands.

```
Brocade(config)#ipv6 router vrrp-extended
Brocade(config-ipv6-VRRP-E-router)#slow-start 60
```

Syntax: [no] slow-start *seconds*

The *seconds* variable specifies a value from 1 through 255.

If the Master subsequently comes back up again, the amount of time specified by the VRRP-E slow start timer elapses (in the IPv4 example, 30 seconds) before the Master takes over from the Backup.

The VRRP-E slow start timer is effective only if the VRRP-E Backup router detects another VRRP-E Master (Standby) router. It is not effective during the initial bootup. The slow start timer is effective on a Backup router if the priority of the Backup router is equal to the configured priority on the Backup state router.

NOTE
The VRRP-E slow start timer applies only to VRRP-E configurations. It does not apply to VRRP configurations.

VRRP-E Extension for Server Virtualization

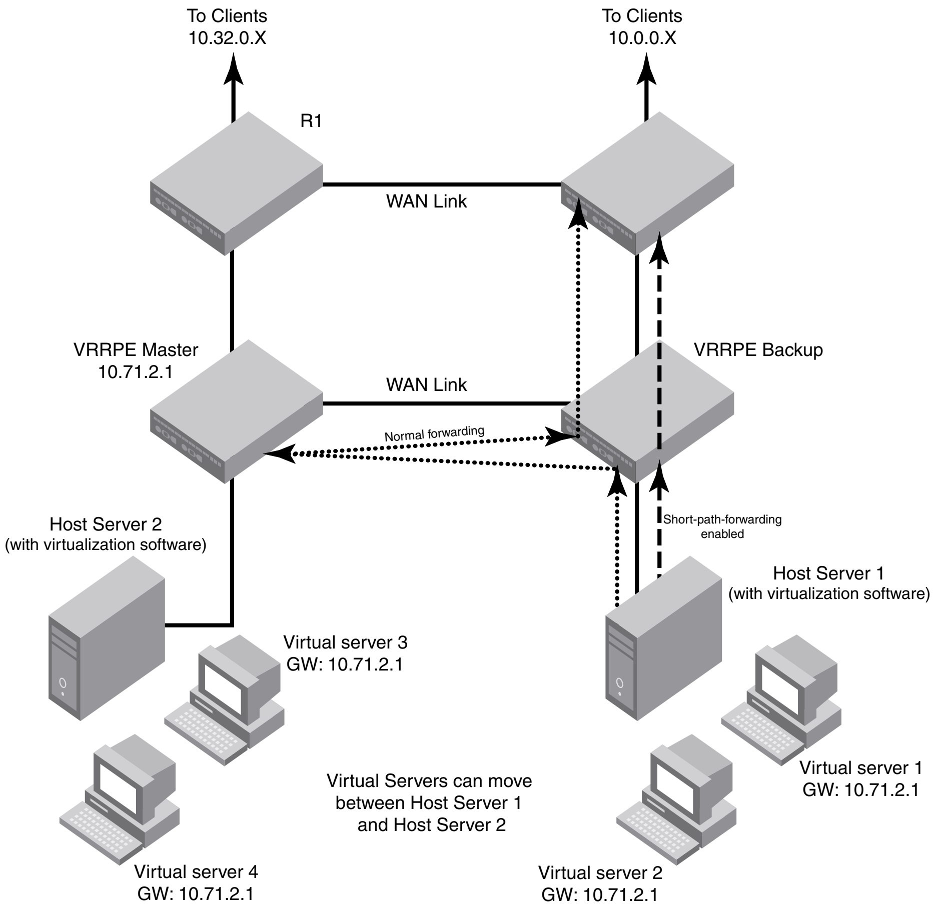
VRRP-E is enhanced with the VRRP-E Extension for Server Virtualization feature so that the Brocade device attempts to bypass the VRRP-E Master router and directly forward packets to their destinations through interfaces on the Backup router.

Figure 33 shows an example of VRRP-E Extension for Server Virtualization. As shown, the virtual servers are dynamically moved between Host Server 1 and Host Server 2. Each time the virtual server is activated, it can be on a different Host Server, and sometimes the traffic crosses the WAN two times before it reaches the client. For example, in the VRRP-E implementation (without VRRP-E Extension for Server Virtualization), traffic from Virtual server 1 to the client at 10.0.0.X was switched to the VRRP-E Master router, then routed back to the VRRP-E Backup router, and then routed to the client (the normal forwarding path).

Short-path forwarding configuration notes

- The VRRP-E Master router and Backup router must have routes to all destinations. You should utilize dynamic routing protocols such as Open Shortest Path First (OSPF) on all routers; otherwise, you must configure the static routes.
- Although it is not required, it is recommended that interfaces on different routers with the same VRID have the same SPF configuration. This ensures that the SPF behavior is retained after a failover. Different VRIDs, however, can have different SPF configurations.

FIGURE 33 VRRP-E Extension for short-path forwarding



VRRP-E Extension for short-path forwarding example

Under the VRRP-E VRID configuration level, there is an option to enable short-path forwarding. To enable short-path forwarding, enter the following commands.

```
Brocade (config)# router vrrp-extended
Brocade (config)# interface ve 10
Brocade (config-vif-10)# ip-address 10.10.10.25/24
Brocade (config-vif-10)# ip vrrp-extended vrid 10
Brocade (config-vif-10-vrid-10)# backup priority 50
Brocade (config-vif-10-vrid-10)# ip-address 10.10.10.254
Brocade (config-vif-10-vrid-10)# short-path-forwarding
Brocade (config-vif-10-vrid-10)# activate
```

Syntax: [no] short-path-forwarding [revert-priority *value*]

The **revert-priority** *value* parameter uses the priority value as the threshold to determine whether the short-path forwarding (SPF) behavior is effective. Typically, when short-path forwarding is enabled, the Backup router enforces SPF. For each port that goes down, the current priority of the VRRP-E router is lowered by the number specified in the **track-port** command. When the current priority is lower than the threshold, the SPF behavior is temporarily suspended and reverts back to the pre-SPF VRRP-E forwarding behavior. The value range is from 1 through 255.

Displaying short-path forwarding combinations

When short-path forwarding (SPF) is configured, the output of the following show commands include the SPF information:

- **show run**
- **show ip vrrp-e brief**
- **show ip vrrp-e vrid** *vrid*

The following example displays information about VRID 1 when only short-path forwarding is configured.

```
Brocade# show ip vrrp-e vrid 1
VRID 1
  Interface ethernet v100
  state backup
  administrative-status enabled
  priority 110
  current priority 90
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3500 msec
  preempt-mode true
  virtual ip address 10.1.1.3
  virtual mac address 02e0.5289.7001
  advertise backup: disabled
  master router 10.1.1.1 expires in 00:00:02.6
  track-port 1/1/13(down)
  short-path-forwarding  enabled
```

The following example displays information about VRID 1 when short-path forwarding and revert-priority are configured.

```
Brocade# show ip vrrp-e vrid 1
VRID 1
  Interface ethernet v100
  state backup
  administrative-status enabled
  priority 110
  current priority 90
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3500 msec
  preempt-mode true
  virtual ip address 10.1.1.3
  virtual mac address 0000.0089.7001
  advertise backup: disabled
  master router 10.1.1.1 expires in 00:00:02.7
  track-port 1/1/13(down)
  short-path-forwarding  enabled  <revertible priority 80  not reverted >
```

Forcing a Master router to abdicate to a Backup router

NOTE
Forcing a Master router to abdicate to a Backup router is not supported for IPv6 VRRP, IPv4 VRRP-E, and IPv6 VRRP-E. It is only supported for IPv4 VRRP.

You can force a VRRP Master to abdicate (give away control) of a VRID to a Backup router by temporarily changing the Master priority to a value less than that of the Backup router.

The VRRP Owner always has priority 255. You can use this feature to temporarily change the Owner priority to a value from 1 through 254.

NOTE
When you change the VRRP Owner priority, the change takes effect only for the current power cycle. The change is not saved to the startup-config file when you save the configuration and is not retained across a reload or reboot. Following a reload or reboot, the VRRP Owner again has priority 255.

To change the Master priority, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1/6
Brocade(config-if-e10000-1/1/6)# ip vrrp vrid 1
Brocade(config-if-e10000-1/1/6-vrid-1)# owner priority 99
```

Syntax: [**no**] **owner priority** *num*

The *num* variable specifies the new priority and can be a number from 1 through 254.

When the command is enabled, the software changes the priority of the Master to the specified priority. If the new priority is lower than at least one Backup priority for the same VRID, the Backup router takes over and becomes the new Master until the next software reload or system reset.

To verify the change, enter the following command from any level of the CLI.

```
Brocade#show ip vrrp
Total number of VRRP routers defined: 1
Interface ethernet v3
auth-type simple text password
VRID 3
  state backup
  administrative-status enabled
  mode non-owner(backup)
  priority 110
  current priority 110
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3500 msec
  preempt-mode true
  ip-address 192.168.3.1
  virtual mac address 0000.0000.0103
  advertise backup: enabled
  next hello sent in 00:00:26.1
  master router 192.168.3.1 expires in 00:00:02.7
  track-port 1/1/1-1/1/4(up)
```

This example shows that even though this Layer 3 switch is the Owner of the VRID (“mode owner”), the Layer 3 switch priority for the VRID is 110 and the state is now “backup” instead of “active”. In addition, the administrative status is “enabled”.

To change the Master priority back to the default Owner priority 255, enter **no** followed by the command you entered to change the priority. For example, to change the priority of a VRRP Owner back to 255 from 110, enter the following command.

```
Brocade(config-if-e10000-1/1/6-vrid-1)#no owner priority 110
```

You cannot set the priority to 255 using the **owner priority** command.

Displaying VRRP and VRRP-E information

You can display the following information for VRRP or VRRP-E:

- Summary configuration and status information
- Detailed configuration and status information
- VRRP and VRRP-E statistics
- CPU utilization statistics

Displaying summary information

To display summary information for a Layer 3 switch for VRRP, enter the **show ip vrrp brief** command at any level of the CLI.

```
Brocade#show ip vrrp brief

Total number of VRRP routers defined: 1
Interface VRID CurPri P State  Master addr   Backup addr       VIP
 1/1/6        1    255  P Init   192.163.5.1    192.168.5.3 192.168.5.1
```

To display summary information for IPv6 VRRP, enter the **show ipv6 vrrp brief** command at any level of the CLI.

```
Brocade#show ipv6 vrrp brief
Total number of VRRP routers defined: 1
Interface     VRID CurPri P State  Master addr
                                Backup addr
                                VIP
1/1/5           1    255    P Master Master addr: Local
                                   Backup addr: 2001:db8:212:f2ff:fea8:3900
                                   VIP        : 2001:db8::1
```

To display summary information for IPv6 VRRP-E v3 , enter the **show ipv6 vrrp-extended brief** command at any level of the CLI.

```
Brocade#show ipv6 vrrp-extended brief
Total number of VRRP-Extended routers defined: 3
Interface     VRID CurPri P State  Master addr
                                Backup addr
                                VIP
1/1/1        1   100   P Master Master addr: Local
                                   Backup addr: 2001:db8:212:f2ff:fea8:5b00
                             VIP       : 2001:db8:1::100
1/1/2        2   150   P Master Master addr: Local
                                   Backup addr: 2001:db8:212:f2ff:fea8:5b00
                             VIP       : 2001:db8:2::100
v51         100  100   P Master Master addr: Local
                                   Backup addr: 2001:db8:212:f2ff:fea8:5b00
                             VIP       : 2001:db8:51::100
```

Syntax for IPv4 VRRP v2 and IPv6 VRRP v3:

Syntax: show ip vrrp brief | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat** | **vrid** *num*

Syntax: show ipv6 vrrp brief | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat** | **vrid** *num*

Syntax for IPv4 VRRP-E v2 and IPv6 VRRP-E v3:

Syntax: show ip vrrp-extended brief | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat** | **vrid** *num*

Syntax: show ipv6 vrrp-extended brief | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat** | **vrid** *num*

The **brief** option displays the summary information. If you do not use this option, detailed information is displayed instead. Refer to "Displaying detailed information" on page 448.

The **ethernet** *slotnum* option is required on chassis devices if you specify a port number.

The **ethernet** *portnum* option specifies an Ethernet port. If you use this option, the command displays VRRP or VRRP-E information only for the specified port.

The **ve** *num* option specifies a virtual interface. If you use this option, the command displays VRRP or VRRP-E information only for the specified virtual interface.

The **stat** option displays statistics. Refer to "Displaying statistics" on page 454.

The **vrid** *num* option specifies the virtual router ID. Enter a value from 1 through 255.

Table 81 shows a description of the output for the **show ip vrrp brief** and **show ip vrrp-extended brief** commands.

TABLE 81 CLI display of VRRP or VRRP-E summary information

Field	Description
Total number of VRRP (or VRRP-Extended) routers defined	The total number of VRIDs configured on this Layer 3 switch. **NOTE:** The total applies only to the protocol the Layer 3 switch is running. For example, if the Layer 3 switch is running VRRP-E, the total applies only to VRRP-E routers.
Interface	The interface on which VRRP or VRRP-E is configured. If VRRP or VRRP-E is configured on multiple interfaces, information for each interface is listed separately.

TABLE 81 CLI display of VRRP or VRRP-E summary information (Continued)

Field	Description
VRID	The VRID configured on this interface. If multiple VRIDs are configured on the interface, information for each VRID is listed in a separate row.
CurPri	The current VRRP or VRRP-E priority of this Layer 3 switch for the VRID.
P	Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains a "P". If the mode is disabled, this field is blank.
State	This Layer 3 switch VRRP or VRRP-E state for the VRID. The state can be one of the following: • Init – The VRID is not enabled (activated). If the state remains Init after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. **NOTE:** If the state is Init and the mode is incomplete, make sure you have specified the IP address for the VRID. • Backup – This Layer 3 switch is a Backup for the VRID. • Master – This Layer 3 switch is the Master for the VRID.
Master addr	IP address of the router interface that is currently Master for the VRID.
Backup addr	IP addresses of router interfaces that are currently Backups for the VRID.
VIP	The virtual IP address that is being backed up by the VRID.

Displaying detailed information

To display detailed VRRP or VRRP-E information, enter the **show ip vrrp** command at any level of the CLI.

```
Brocade#show ip vrrp

Total number of VRRP routers defined: 1
Interface ethernet v3
auth-type simple text password
VRID 3
  state master
  administrative-status enabled
  mode owner
  priority 255
  current priority 255
  track-priority 150
  hello-interval 1000 msec
  ip-address 192.168.3.1
  virtual mac address 0000.5e00.0103
  advertise backup: disabled
  next hello sent in 00:00:00.7
  backup router 192.168.3.2 expires in 00:02:41.3
  track-port 1/1/14(up)
```

The following example is for a VRRP Backup.

```
Brocade#show ip vrrp

Total number of VRRP routers defined: 1
Interface ethernet v3
auth-type simple text password
VRID 3
  state backup
  administrative-status enabled
  mode non-owner(backup)
  priority 110
  current priority 110
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3500 msec
  preempt-mode true
  ip-address 192.168.3.1
  virtual mac address 0000.0000.0103
  advertise backup: enabled
  next hello sent in 00:00:26.1
  master router 192.168.3.1 expires in 00:00:02.7
  track-port 1/2/1-1/2/4(up)
```

The following example is for a VRRP-E Master.

```
Brocade#show ip vrrp-extended

Total number of VRRP-Extended routers defined: 50
Interface ethernet v201
 auth-type simple text password
 VRID 201
  state master
  administrative-status enabled
  priority 220
  current priority 220
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3100 msec
  preempt-mode true
  virtual ip address 10.201.201.5
  virtual mac address 0000.00d7.82c9
  advertise backup: enabled
  next hello sent in 00:00:00.1
  backup router 10.201.201.4 expires in 00:02:45.2
  backup router 10.201.201.3 expires in 00:02:47.6
  track-port 1/1/5*1/1/24(up)
```

Syntax: **show ip vrrp brief** | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat**

Syntax: **show ip vrrp-extended brief** | **ethernet** *stack-unit/slotnum/portnum* | **ve** *num* | **stat**

The **brief** option displays summary information. Refer to "Displaying summary information" on page 446.

The **ethernet** *portnum* option specifies an Ethernet port. If you use this option, the command displays VRRP or VRRP-E information only for the specified port. Also, you must specify the *slotnum* variable on chassis devices.

The **ve** *num* option specifies a virtual interface. If you use this option, the command displays VRRP or VRRP-E information only for the specified virtual interface.

The **stat** option displays statistics. Refer to "Displaying statistics" on page 454.

Table 82 shows a description of the output for the **show ip vrrp** and **show ip vrrp-extended** commands.

TABLE 82 CLI display of VRRP or VRRP-E detailed information

Field	Description
Total number of VRRP (or VRRP-Extended) routers defined	The total number of VRIDs configured on this Layer 3 switch. NOTE: The total applies only to the protocol the Layer 3 switch is running. For example, if the Layer 3 switch is running VRRP-E, the total applies only to VRRP-E routers.
Interface parameters	
Interface	The interface on which VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E is configured. If VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E is configured on multiple interfaces, information for each interface is listed separately.
auth-type	The authentication type enabled on the interface.
VRID parameters	
VRID	The VRID configured on this interface. If multiple VRIDs are configured on the interface, information for each VRID is listed separately.
state	This Layer 3 switch VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E state for the VRID. The state can be one of the following: • initialize – The VRID is not enabled (activated). If the state remains "initialize" after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. NOTE: If the state is "initialize" and the mode is incomplete, make sure you have specified the IP address for the VRID. • backup – This Layer 3 switch is a Backup for the VRID. • master – This Layer 3 switch is the Master for the VRID.
administrative-status	The administrative status of the VRID. The administrative status can be one of the following: • disabled – The VRID is configured on the interface but VRRP or VRRP-E has not been activated on the interface. • enabled – VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E has been activated on the interface.
mode	Indicates whether the Layer 3 switch is the Owner or a Backup for the VRID. NOTE: If "incomplete" appears after the mode, configuration for this VRID is incomplete. For example, you might not have configured the virtual IP address that is being backed up by the VRID. NOTE: This field applies only to VRRP or VRRP v3. All Layer 3 switches configured for VRRP-E are Backups.
priority	The device preferability for becoming the Master for the VRID. During negotiation, the router with the highest priority becomes the Master. If two or more devices are tied with the highest priority, the Backup interface with the highest IP address becomes the active router for the VRID.

TABLE 82 CLI display of VRRP or VRRP-E detailed information (Continued)

Field	Description
current priority	The current VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E priority of this Layer 3 switch for the VRID. The current priority can differ from the configured priority (refer to the priority field) for the following reason: The current priority can differ from the configured priority in the VRID if the VRID is configured with track ports and the link on a tracked interface has gone down. Refer to "Track ports and track priority" on page 416.
hello-interval	The configured value for the Hello interval. This is the amount of time, in milliseconds, between Hello messages from the Master to the Backups for a given VRID. **NOTE:** In some VRRP command outputs, Hello interval timers are displayed in seconds instead of milliseconds.
dead interval	The configured value for the dead interval. This is the amount of time, in milliseconds, that a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID. **NOTE:** If the value is 0, then you have not configured this parameter. **NOTE:** This field does not apply to VRRP Owners. **NOTE:** All timer fields (Hello interval, dead interval, current dead interval, and so on) are displayed in milliseconds.
current dead interval	The current value of the dead interval. This value is equal to the value configured for the dead interval. If the value for the dead interval is not configured, then the current dead interval is equal to three times the Hello interval plus Skew time (where Skew time is equal to 256 minus priority divided by 256). **NOTE:** This field does not apply to VRRP Owners.
preempt mode	Whether the backup preempt mode is enabled. **NOTE:** This field does not apply to VRRP Owners.
virtual ip address	The virtual IP addresses that this VRID is backing up. The address can be an IPv4 or IPv6 address.
virtual mac address	The virtual MAC addresses for the VRID. The MAC address can be an IPv4 or IPv6 address.
advertise backup	The IP addresses of Backups that have advertised themselves to this Layer 3 switch by sending Hello messages. **NOTE:** Hello messages from Backups are disabled by default. You must enable the Hello messages on the Backup for the Backup to advertise itself to the current Master. Refer to "Hello messages" on page 415.
backup router *ip-addr* expires in *time*	The IP addresses of Backups that have advertised themselves to this Master by sending Hello messages. The *time* value indicates how long before the Backup expires. A Backup expires if you disable the advertise backup option on the Backup or the Backup becomes unavailable. Otherwise, the Backup next Hello message arrives before the Backup expires. The Hello message resets the expiration timer. An expired Backup does not necessarily affect the Master. However, if you have not disabled the advertise backup option on the Backup, then the expiration may indicate a problem with the Backup. **NOTE:** This field applies only when Hello messages are enabled on the Backups (using the advertise backup option).

TABLE 82 CLI display of VRRP or VRRP-E detailed information (Continued)

Field	Description
next hello sent in *time*	How long until the Backup sends its next Hello message. **NOTE:** This field applies only when this Layer 3 switch is the Master and the Backup is configured to send Hello messages (the advertise backup option is enabled).
master router *ip-addr* expires in *time*	The IP address of the Master and the amount of time until the Master dead interval expires. If the Backup does not receive a Hello message from the Master by the time the interval expires, either the IP address listed for the Master will change to the IP address of the new Master, or this Layer 3 switch itself will become the Master. **NOTE:** This field applies only when this Layer 3 switch is a Backup.
track port	The interfaces that the VRID interface is tracking. If the link for a tracked interface goes down, the VRRP, VRRP v3, VRRP-E, or IPv6 VRRP-E priority of the VRID interface is changed, causing the devices to renegotiate for Master. **NOTE:** This field is displayed only if track interfaces are configured for this VRID.

Displaying detailed information for an individual VRID

You can display information about the settings configured for a specified VRRP Virtual Router ID (VRID). To display information about VRID 1, enter the **show ip vrrp vrid** command.

```
Brocade#show ip vrrp vrid 2
VRID 2
  Interface ethernet v2
  state master
  administrative-status enabled
  version v2
  mode non-owner(backup)
  priority 100
  current priority 100
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3600 msec
  preempt-mode true
  ip-address 10.1.1.5
  virtual mac address 0000.0000.0102
  advertise backup: disabled
  next hello sent in 00:00:01.0
```

To display information about the settings configured for a specified IPv6 VRRP VRID, enter the **show ipv6 vrrp vrid** command.

```
Brocade#show ipv6 vrrp vrid 1
VRID 1
  Interface ethernet 5
  state backup
  administrative-status enabled
  version v3
  mode non-owner(backup)
  priority 100
  current priority 100
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3000 msec
  preempt-mode true
  ip-address 2001:db8:a7a7::1
  virtual mac address 0000.0000.0201
  advertise backup: enabled
  next hello sent in 00:00:38.0
    track-port 1/1/5(up)
  master router 2001:db8:212:f2ff:fea8:5b00 timer expires in 00:00:02.6
```

Syntax: **show ip vrrp vrid** *num* [**ethernet** *num* | **ve** *num*]

Syntax: **show ipv6 vrrp vrid** *num* [**ethernet** *num* | **ve** *num*]

The *num* variable specifies the VRID.

The **ethernet** *num* | **ve** *num* options specify an interface on which the VRID is configured. If you specify an interface, VRID information is displayed for that interface only. Otherwise, information is displayed for all the interfaces on which the specified VRID is configured.

Table 83 shows a description of the output for the **show ip vrrp vrid** command.

TABLE 83 Output from the show ip vrrp vrid command

Field	Description
VRID	The specified VRID.
Interface	The interface on which VRRP is configured.
State	This Layer 3 switch VRRP state for the VRID. The state can be one of the following: • Init – The VRID is not enabled (activated). If the state remains Init after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. NOTE: If the state is Init and the mode is incomplete, make sure you have specified the IP address for the VRID. • Backup – This Layer 3 switch is a Backup for the VRID. • Master – This Layer 3 switch is the Master for the VRID.
priority	The configured VRRP priority of this Layer 3 switch for the VRID.
current priority	The current VRRP priority of this Layer 3 switch for the VRID.
track priority	The new VRRP priority that the router receives for this VRID if the interface goes down.
hello interval	How often the Master router sends Hello messages to the Backups.
dead interval	The amount of time a Backup waits for a Hello message from the Master before determining that the Master is dead.

TABLE 83 Output from the show ip vrrp vrid command (Continued)

Field	Description
current dead interval	The current value of the dead interval. This value is equal to the value configured for the dead interval. If the value for the dead interval is not configured, then the current dead interval is equal to three times the Hello interval plus Skew time (where Skew time is equal to 256 minus priority divided by 256). **NOTE:** This field does not apply to VRRP Owners.
preempt mode	Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains “true”. If the mode is disabled, this field contains “false”.
advertise backup	Whether Backup routers send Hello messages to the Master.

Displaying statistics

To display statistics on most Brocade devices, enter the **show ip vrrp stat** command at any level of the CLI.

```
Brocade#show ip vrrp stat

Interface ethernet 1/1/5
 rxed vrrp header error count = 0
 rxed vrrp auth error count = 0
 rxed vrrp auth passwd mismatch error count = 0
 rxed vrrp vrid not found error count = 0
 VRID 1
 rxed arp packet drop count = 0
 rxed ip packet drop count = 0
 rxed vrrp port mismatch count = 0
 rxed vrrp ip address mismatch count = 0
 rxed vrrp hello interval mismatch count = 0
 rxed vrrp priority zero from master count = 0
 rxed vrrp higher priority count = 0
 transitioned to master state count = 1
 transitioned to backup state count = 1
```

To display IPv6 VRRP-E v3 statistics on a device, enter the following command at any level of the CLI.

```
 Brocade#show ipv6 vrrp-extended stat ve 51
 Interface ethernet v51
  rxed vrrp header error count = 0
  rxed vrrp auth error count = 0
  rxed vrrp auth passwd mismatch error count = 0
  rxed vrrp vrid not found error count = 0
  VRID 100
  rxed arp packet drop count = 0
  rxed ip packet drop count = 0
  rxed vrrp port mismatch count = 0
  rxed vrrp ip address mismatch count = 0
  rxed vrrp hello interval mismatch count = 0
  rxed vrrp priority zero from master count = 0
  rxed vrrp higher priority count = 0
```

Table 84 shows a description of the output for the **show ip vrrp stat** and **show ip vrrp- extended stat** commands.

TABLE 84 CLI display of VRRP or VRRP-E statistics

Field	Description
Interface statistics	
Interface	The interface on which VRRP or VRRP-E is configured. If VRRP or VRRP-E is configured on more than one interface, the display lists the statistics separately for each interface.
rxed vrrp header error count	The number of VRRP or VRRP-E packets received by the interface that had a header error.
rxed vrrp auth error count	The number of VRRP or VRRP-E packets received by the interface that had an authentication error.
rxed vrrp auth passwd mismatch error count	The number of VRRP or VRRP-E packets received by the interface that had a password value that does not match the password used by the interface for authentication.
rxed vrrp vrid not found error count	The number of VRRP or VRRP-E packets received by the interface that contained a VRID that is not configured on this interface.
VRID statistics	
rxed arp packet drop count	The number of ARP packets addressed to the VRID that were dropped.
rxed ip packet drop count	The number of IP packets addressed to the VRID that were dropped.
rxed vrrp port mismatch count	The number of packets received that did not match the configuration for the receiving interface.
rxed vrrp ip address mismatch count	The number of packets received that did not match the configured IP addresses.
rxed vrrp hello interval mismatch count	The number of packets received that did not match the configured Hello interval.
rxed vrrp priority zero from master count	Indicates that the current Master has resigned.
rxed vrrp higher priority count	The number of VRRP or VRRP-E packets received by the interface that had a higher backup priority for the VRID than this Layer 3 switch backup priority for the VRID.
transitioned to master state count	The number of times this Layer 3 switch has changed from the backup state to the master state for the VRID.
transitioned to backup state count	The number of times this Layer 3 switch has changed from the master state to the backup state for the VRID.

Clearing VRRP or VRRP-E statistics

To clear VRRP or VRRP-E statistics, enter the **clear ip vrrp-stat** command at the Privileged EXEC level or any configuration level of the CLI.

```
Brocade#clear ip vrrp-stat
```

Syntax: **clear ip vrrp-stat**

To clear IPv6 VRRP v3 or IPv6 VRRP-E v3 statistics, enter the following command at the Privileged EXEC level or any configuration level of the CLI.

```
Brocade#clear ipv6 vrrp-stat
```

Syntax: **clear ipv6 vrrp-stat**

Displaying CPU utilization statistics

To display CPU utilization statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI. In the following output example, IPv6 protocols are also displayed.

```
Brocade#show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.02      0.02      0.02      0.02          15532
BGP              0.00      0.00      0.00      0.00              0
DOT1X            0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.01      0.01      0.01           8608
IP               0.09      0.12      0.11      0.08          72959
OSPF             0.05      0.07      0.07      0.06          85015
RIP              0.00      0.00      0.00      0.00             98
STP              0.68      0.86      0.78      0.57         568586
VRRP             0.42      0.54      0.50      0.37         357133

Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
IPv6             1.23      1.49      0.99      1.40         917973
ICMP6            0.04      0.05      0.06      0.04          38508
ND6              0.00      0.01      0.01      0.01           6691
RIPng            0.00      0.00      0.00      0.00             45
OSPFv3           0.00      0.00      0.00      0.00           1515
IPV6_RX          0.16      0.21      0.21      0.14         143506
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running, as shown in the following example.

```
Brocade#show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00              0
BGP              0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.00      0.00      0.00              1
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following. In the following output example, IPv6 protocols are also displayed for a specific number of seconds.

```
Brocade#show process cpu 2
Statistics for last 1 sec and 999 ms
Process Name   Sec(%)   Time(ms)
ARP              0.01         0
BGP              0.00         0
DOT1X            0.00         0
GVRP             0.00         0
ICMP             0.01         0
IP               0.04         0
OSPF             0.07         1
RIP              0.00         0
STP              0.97        19
VRRP             0.53        10

Statistics for last 1 sec and 999 ms
Process Name   Sec(%)   Time(ms)
IPv6             0.09         1
ICMP6            0.05         1
ND6              0.00         0
RIPng            0.00         0
OSPFv3           0.00         0
IPV6_RX          0.04         0
```

When you specify how many seconds of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is for the previous 1 second plus 80 milliseconds.

Syntax: **show process cpu** [*num*]

The *num* variable specifies the number of seconds and can be a value from 1 through 900. If you use this variable, the command lists the usage statistics only for the specified number of seconds. If you do not use this variable, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying VRRP and VRRP-E information for IPv6

You can display information for IPv6 VRRP or VRRP-E v3.

Displaying detailed information for IPv6 VRRP v3 and IPv6 VRRP-E v3

To display information for an IPv6 VRRP Owner, enter the **show ipv6 vrrp** command at any level of the CLI.

```
Brocade#show ipv6 vrrp
Total number of VRRP routers defined: 25
Interface ethernet v52
auth-type no authentication
VRID 52
  state master
  administrative-status enabled
  version v3
  mode owner
  priority 255
  current priority 255
  track-priority 5
  hello-interval 1000 msec
  ipv6-address 2001:db8::52:3
  virtual mac address 0000.0000.0234
  advertise backup: disabled
  next hello sent in 00:00:00.1
  backup router 2001:db8:224:38ff:fec8:5a40 expires in 00:02:03.1
```

To display information for an IPv6 VRRP Backup, enter the **show ipv6 vrrp** command at any level of the CLI.

```
Brocade#show ipv6 vrrp
Total number of VRRP routers defined: 26
Interface ethernet v52
auth-type no authentication
 VRID 52
  state backup
  administrative-status enabled
  version v3
  mode non-owner(backup)
  priority 101
  current priority 20
  track-priority 20
  hello-interval 100 msec
  dead-interval 0 msec
  current dead-interval 300 msec
  preempt-mode true
  ipv6-address 2001:db8::52:3
  virtual mac address 0000.0000.0234
  advertise backup: enabled
  next hello sent in 00:00:36.5
  master router 2001:db8:768e:f8ff:fe33:8600 expires in 00:00:00.2
  track-port 1/1/3*1/1/8(down) v41(up)
```

Syntax: **show ipv6 vrrp brief** | **ethernet** *stack-unit/slotnum/portnum* | **stat** [**ethernet** *stack-unit/slotnum/portnum* | **ve** *num*] | **vrid** *num*

To display detailed information for IPv6 VRRP-E, enter the **show ipv6 vrrp-extended** command at any level of the CLI.

```
Brocade#show ipv6 vrrp-extended
Total number of VRRP-Extended routers defined: 1
Interface ethernet v201
auth-type md5 authentication
VRID 201
  state master
  administrative-status enabled
  priority 100
  current priority 100
  hello-interval 1000 msec
  dead-interval 0 msec
  current dead-interval 3600 msec
  preempt-mode true
  virtual ipv6 address 2001:db8::201:5
  virtual mac address 0200.0002.bac9
  advertise backup: enabled
  next hello sent in 00:00:01.0
```

Syntax: **show ipv6 vrrp-extended brief** | **ethernet** *stack-unit/slotnum/portnum* | **stat** [**ethernet** *stack-unit/slotnum/portnum* | **ve** *num*] | **vrid** *num*

For more information on the field descriptions for the **show ipv6 vrrp** command and the **show ipv6 vrrp -extended** command, refer to "CLI display of VRRP or VRRP-E detailed information" on page 450.

Configuration examples

The following sections contain the CLI commands for implementing the VRRP and VRRP-E configurations shown in Figure 31 on page 413 and Figure 32 on page 419.

VRRP example

To implement the VRRP configuration shown in Figure 31 on page 413, use the following method.

Configuring Switch 1

To configure VRRP Switch 1, enter the following commands.

```
Brocade Switch1(config)#router vrrp
Brocade Switch1(config)#interface ethernet 1/1/6
Brocade Switch1(config-if-e10000-1/1/6)#ip-address 192.168.5.1/24
Brocade Switch1(config-if-e10000-1/1/6)#ip vrrp vrid 1
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#owner track-priority 20
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#track-port ethernet 1/1/2
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#ip-address 192.168.5.1
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#activate
```

NOTE
When you configure the Master (Owner), the address you enter with the **ip-address** command must already be configured on the interface.

Configuring Switch 2

To configure Switch 2 in Figure 31 on page 413 after enabling VRRP, enter the following commands.

```
Brocade Switch2(config)#router vrrp
Brocade Switch2(config)#interface ethernet 1/1/5
Brocade Switch2(config-if-e10000-1/1/5)#ip-address 192.168.5.3/24
Brocade Switch2(config-if-e10000-1/1/5)#ip vrrp vrid 1
Brocade Switch2(config-if-e10000-1/1/5-vrid-1)#backup priority 100
track-priority 19
Brocade Switch2(config-if-e10000-1/1/5-vrid-1)#track-port ethernet 1/1/3
Brocade Switch2(config-if-e10000-1/1/5-vrid-1)#ip-address 192.168.5.1
Brocade Switch2(config-if-e10000-1/1/5-vrid-1)#activate
```

The **backup** command specifies that this router is a VRRP Backup for virtual router VRID1. The IP address entered with the **ip-address** command is the same IP address as the one entered when configuring Switch 1. In this case, the IP address cannot also exist on Switch 2, but the interface on which you are configuring the VRID Backup must have an IP address in the same subnet. By entering the same IP address as the one associated with this VRID on the Owner, you are configuring the Backup to back up the address, but you are not duplicating the address.

NOTE
When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same subnet as the address associated with the VRID by the Owner. However, the address cannot be the same.

The **priority** parameter establishes the router VRRP priority in relation to the other VRRP routers in this virtual router. The **track-priority** parameter specifies the new VRRP priority that the router receives for this VRID if the interface goes down. Refer to "Track ports and track priority" on page 416.

The **activate** command activates the VRID configuration on this interface. The interface does not provide backup service for the virtual IP address until you activate the VRRP configuration. Alternatively, you can use the **enable** command. The **activate** and **enable** commands do the same thing.

Syntax: **router vrrp**

Syntax: **ip vrrp vrid** *vrid*

Syntax: **owner** [**track-priority** *value*]

Syntax: **backup** [**priority** *value*] [**track-priority** *value*]

Syntax: **track-port ethernet** *stack-unit/slotnum/portnum* | **ve** *num*

Syntax: **ip-address** *ip-addr*

Syntax: **activate**

VRRP-E example

To implement the VRRP-E configuration shown in Figure 32 on page 419, use the following CLI method.

Configuring Switch 1

To configure VRRP Switch 1 in Figure 32 on page 419, enter the following commands.

```
Brocade Switch1(config)#router vrrp-extended
Brocade Switch1(config)#interface ethernet 1/1/6
Brocade Switch1(config-if-e10000-1/1/6)#ip-address 192.168.5.2/24
Brocade Switch1(config-if-e10000-1/1/6)#ip vrrp-extended vrid 1
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#backup priority 110 track-priority
20
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#track-port ethernet 1/1/4
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#ip-address 192.168.5.254
Brocade Switch1(config-if-e10000-1/1/6-vrid-1)#activate
VRRP router 1 for this interface is activating
Brocade Switch1(config-if-e10000-1/1/6)#
Brocade-Switch1(config-if-e10000-1/1/6)#ip vrrp-e vrid 2
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#backup priority 100 track-priority
20
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#track-port ethernet 1/1/4
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#ip-address 192.168.5.253
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#activate
VRRPE router 2 for this interface is activating
```

NOTE
The address you enter with the **ip-address** command cannot be the same as a real IP address configured on the interface.

Configuring Switch 2

To configure Switch 2, enter the following commands.

```
Brocade-Switch1(config)#router vrrp-extended
Brocade-Switch1(config-vrrpe-router)#interface ethernet 1/2/1
Brocade-Switch1(config-if-e10000-1/1/6)#ip address 192.168.5.3/24
Brocade-Switch1(config-if-e10000-1/1/6)#ip vrrp-extended vrid 1
Brocade-Switch1(config-if-e10000-1/1/6-vrid-1)#backup priority 100 track-priority 
20
Brocade-Switch1(config-if-e10000-1/1/6-vrid-1)#track-port ethernet 1/1/3
Brocade-Switch1(config-if-e10000-1/1/6-vrid-1)#ip-address 192.168.5.254
Brocade-Switch1(config-if-e10000-1/1/6-vrid-1)#activate
VRRPE router 1 for this interface is activating
Brocade-Switch1(config-if-e10000-1/1/6)#
Brocade-Switch1(config-if-e10000-1/1/6)#ip vrrp-e vrid 2
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#backup priority 110 track-priority 
20
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#track-port ethernet 1/1/3
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#ip-address 192.168.5.253
Brocade-Switch1(config-if-e10000-1/1/6-vrid-2)#activate
VRRPE router 2 for this interface is activating
```

The **backup** command specifies that this router is a VRRP-E Backup for virtual router VRID1. The IP address entered with the **ip-address** command is the same IP address as the one entered when configuring Switch 1. In this case, the IP address cannot also exist on Switch 2, but the interface on which you are configuring the VRID Backup must have an IP address in the same subnet. By entering the same IP address as the one associated with this VRID on the Owner, you are configuring the Backup to back up the address, but you are not duplicating the address.

NOTE
When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same subnet as the address associated with the VRID by the Owner. However, the address cannot be the same.

The **priority** parameter establishes the router VRRP-E priority in relation to the other VRRP-E routers in this virtual router. The **track-priority** parameter specifies the new VRRP-E priority that the router receives for this VRID if the interface goes down. Refer to “Track ports and track priority” on page 416.

The **activate** command activates the VRID configuration on this interface. The interface does not provide backup service for the virtual IP address until you activate the VRRP-E configuration. Alternatively, you can use the **enable** command. The **activate** and **enable** commands do the same thing.

Syntax: **router vrrp-extended**

Syntax: **ip vrrp-extended vrid** *vrid*

Syntax: **backup** [**priority** *value*] [**track-priority** *value*]

Syntax: **track-port ethernet** *stack-unit/slotnum/portnum* | **ve** *num*

Syntax: **ip-address** *ip-addr*

Syntax: **activate**

Index